REFERENCE GUIDE
DI
AMIGA MAGAZINE

a cura della redazione di Amiga Magazine
in collaborazione con GVP
© 1992 - 1993 Gruppo Editoriale Jackson

GRUPPO EDITORIALE
JACKSON

© Copyright per l'edizione originale:
Gruppo Editoriale Jackson 1992, 1993

Coordinamento Tecnico e Redazionale:
Massimiliano Anticoli
Redazione:
Daniele Cassanelli, Domenico Pavone,
Romano Tenca
Grafica e impaginazione elettronica:
DTP Studio di Piera Loddo

Tutti i diritti sono riservati. Stampato in Italia.
Nessuna parte di questo libro può essere riprodotta, memorizzata in sistemi di archivio, o trasmessa in qualsiasi forma o mezzo elettronico, fotocopia, registrazione o altri senza la preventiva autorizzazione scritta dell'editore.

Sommario

Capitolo 1

Introduzione ad Amiga

Capitolo 2

Hardware

Capitolo 3

Grafica

Capitolo 4

Audio

Capitolo 5

L'Exec

Capitolo 6

AmigaDos 1.3: il CLI

Capitolo 7

Il Workbench

Capitolo 8

Il Workbench 2.0

Capitolo 9

AmigaDos 2.0: l'ambiente Shell

Capitolo 10

Il Workbench 3.0: ultime novità

Capitolo 11

Non solo games

Capitolo 1

INTRODUZIONE AD AMIGA

Storia di Amiga

Amiga viene alla luce nel lontano 1984; ci si rende subito conto che non si tratta di un semplice miglioramento del 64, l'home computer che ha reso famosa la Commodore, ma che si ha a che fare con un calcolatore completamente nuovo e rivoluzionario grazie alla sua struttura interna. Grazie ai coprocessori, ai chip custom e ai 25 canali DMA, Amiga si impone immediatamente come il personal computer ad avere il miglior rapporto qualità-prezzo.

Modelli disponibili

Fino a oggi, Amiga ha subito innumerevoli ritocchi sia dal punto di vista hardware che software. Da quel "preistorico" ***Amiga 1000*** con 256 Kb di RAM e sistema operativo 1.0 su disco ne è passata di acqua sotto i ponti. Allora il sistema operativo veniva caricato da disco e rimaneva residente in memoria, in una zona protetta dalla scrittura. Successivamente A1000 è stato commercializzato con 512 Kb di memoria, ma i grandi bug del sistema operativo rimanevano, visto che i Guru Meditation erano molto frequenti. All'epoca la stampa del settore era molto critica nei confronti di Amiga, non tanto per la sua struttura interna, ma per gli innumerevoli bachi del sistema operativo assolutamente inaffi-

dabile anche per brevi lavori; era inconcepibile che una macchina che potenzialmente poteva competere con sistemi professionali, staccando di gran lunga i PC IBM e gli Apple Macintosh, avesse queste pesanti limitazioni, nonostante il rilascio della release 1.1 del Kickstart.
Il grande passo avanti viene effettuato con la versione 1.2 del sistema operativo (siamo nel 1987), sensibilmente più solido e affidabile dei suoi predecessori, e con la commercializzazione di due nuovi modelli, ***Amiga 500*** e ***Amiga 2000***. La novità più importante di questi due nuovi calcolatori è la presenza del Kickstart (ovviamente 1.2) su ROM che evita il caricamento del disco iniziale.
I due computer sono però destinati a utenti diversi: A500 vuole entrare nelle case di chi non ha intenzione di impiegarlo come macchina da lavoro, mentre A2000 è destinato a coloro che lo vogliono utilizzare professionalmente; lo testimoniano la presenza di numerosi slot d'espansione, alcuni dei quali destinati ad una scheda di emulazione IBM, gli altri per l'espansione del sistema con ulteriore memoria o l'installazione di un Hard Disk, l'alloggiamento per due drive, uno slot CPU per future schede acceleratrici e uno slot video.
Nel 1988 la Commodore rilascia la release 1.3 del sistema operativo che sembra essere la versione definitiva; si dimezzano i Guru Meditation e contemporaneamente iniziano le commercializzazioni di schede di espansione della memoria, delle prime schede acceleratrici e dei primissimi controller per hard disk.
Siamo nel 1990 quando, accompagnata da insistenti voci di corridoio, viene presentato ufficialmente Amiga 3000, un calcolatore dalle prestazioni eccezionali, e che , come avremo modo di vedere dalle caratteristiche interne, non ha nulla da invidiare alle workstation grafiche. Le novità riguardano innanzitutto il microprocessore che non è più l'ormai obsoleto MC68000 ma è il ***MC68030*** (a 16 o 25 MHz) affiancato dal coprocessore matematico ***MC68882***. Vengono aggiornati i chip custom (ECS, Enhanced Chip Set), che possono indirizzare fino 2 Mb di Chip RAM (A3000 è dotato di 1 Mb di Chip RAM e 1 di Fast RAM) e viene inserito sulla scheda madre un controller SCSI per hard disk (A3000 è dotato di HD da 50 o 100 Mb). E, come succede nella maggior parte dei casi, l'uscita del nuovo computer è accompagnata dal rilascio della

nuova versione del sistema operativo: non la 1.4, come si vociferava da parecchio tempo, ma la 2.0, a indicare il grande salto di qualità rispetto alla precedente 1.3.
Siamo giunti ai nostri giorni; l'O.S. 2.0 è divenuto ormai realtà e all'A3000 vengono affiancati due nuovi modelli: l'***A500 Plus***, dotato appunto di O.S. 2.0, 1 Mb di Chip Ram (espandibile a 2 Mb), ECS e la versione ***Tower*** di Amiga 3000, identico nella struttura interna, ma dotato di una configurazione di base migliore (MC 68030 a 25 MHz, 5 Mb di RAM, hard disk da 100 Mb); al momento in cui scriviamo i modelli di Amiga ssono quelli appena descritti, ma sicuramente l'evoluzione di Amiga non si ferma qui. Non ci stupiremmo se sentissimo l'annucio di un Amiga dotato di 68040 (tra le altre cose, sul mercato sono già disponibili da tempo schede acceleratrici di questo tipo) che potrebbe chiamarsi ***Amiga 4000*** o magari un ***Amiga portatile*** visto che proprio in questi mesi è iniziata la commercializzazione di portatili a colori.

Struttura interna

La potenza di Amiga è dovuta a una struttura interna di moderna progettazione che lo porta ai vertici della classifica dei personal computer. In effetti Amiga, quando debuttò, era l'unico computer ad avere la possibilità di effettuare un reale multitasking o visualizzare contemporaneamente ben 4096 colori.
Per tutta questa potenza dobbiamo ringraziare i 25 canali DMA, il processore centrale, un Motorola ***MC68000***, che nelle sue funzioni è coadiuvato da tre chip custom, appositamente progettati e realizzati su misura per Amiga: ***Agnus***, ***Denise*** e ***Paula***.
Nei computer normali (i PC IBM per intenderci) se una periferica, per esempio, deve scrivere in memoria i suoi dati passano per forza dalla CPU che li riversa poi in memoria. Amiga, invece, grazie ai canali DMA, rende possibile un trasferimento diretto tra periferica e memoria in modo che il microprocessore sia libero di effettuare calcoli e operazioni più complesse. Su Amiga potremo sentire una musica a quattro voci mentre stiamo ammirando un'immagine a 4096 colori, caricando un programma da hard disk e stampando i nostri documenti.

Central Processing Unit

Il microprocessore 68000 costituisce il cuore di Amiga. È grazie alla sua struttura interna che Amiga riesce a elaborare più programmi contemporaneamente. Il 68000 è un microprocessore progettato per gestire sistemi multiutente; oltre a essere veloce è in grado di eseguire contemporaneamente più programmi soddisfacendo tramite una ben precisa gerarchia (interrupt ed exception) le richieste che provengono dai singoli processi. Il 68000 ha due diverse modi di elaborazione: uno in modalità ***utente***, che corrisponde a quello sopra descritto (la CPU concede a ogni processo un determinato intervallo di tempo per elaborare i singoli programmi), e l'altro in modalità ***supervisore***, dove è possibile gestire il microprocessore nelle sue complete funzioni.
Se a queste potenti funzioni uniamo coprocessori, canali DMA e un ottimo sistema operativo, otteniamo una macchina veloce che distanzia di gran lunga le sue concorrenti. La sua velocità, oltre che dalla struttura della macchina, è da basata sull'architettura del microprocessore; infatti il 68000 è un processore a 16/32 bit: ciò significa che comunica con il "mondo esterno" inviando dati a blocchi di 16 bit, mentre internamente elabora i dati addirittura a 32 bit. Altra peculiarità del 68000 è la quantità di memoria che è in grado di indirizzare; a differenza del bus dati che, come è già stato detto, è a 16 bit e dei bus interni che sono a 32 bit, il bus indirizzi (il canale attraverso il quale si seleziona la cella di memoria che si vuole leggere o scrivere con quelle informazioni che viaggiano sul bus dati) è a 24 bit. Grazie a questi 24 bit si è in grado di accedere a 16.777.216 byte (16 Mb) di memoria contro i 65.536 byte (64 Kb) di un normale processore a 8 bit.

I Chip Custom

I chip custom sono chip progettati e realizzati appositamente per Amiga (custom-built significa letteralmente "costruito su ordinazione") con lo scopo di alleviare il lavoro del microprocessore. I tre chip custom già citati in precedenza sono Agnus, Denise e Paula che affiancano il 68000 svolgendo compiti che, come vedremo in seguito, sugli altri computer sono svolti dalla CPU.

Agnus

Agnus, ossia Address Generator Chip, ha tre funzioni specifiche, che sono:

- controllo del DMA, per permettere alle altri parti che compongono il sistema di non utilizzare la CPU per l'accesso in memoria. I canali DMA hanno il preciso scopo di lasciare libera la CPU di lavorare, soddisfacendo richiesta provenienti da sistemi minori;
- sincronizzazione, tramite il coprocessore copper, dell'emissione dei fasci di elettroni che hanno una frequenza di 50Hz. Questa operazione richiede una rapidità di esecuzione e una precisione che occuperebbero parecchio "tempo macchina"; anche in questo caso Agnus toglie un impegno oneroso alla CPU;
- gestione del blitter (block image transferrer), per lo spostamento rapido di ampie aree di memoria (solitamente a scopo grafico). Lo spostamento di aree di memoria, operazione che fondamentalmente potrebbe svolgere tranquillamente la CPU, potrebbe divenire pesante e occupare troppo la CPU quando queste diventano veloci e le aree di memoria ampie.

Si noti che se le stesse operazioni che compie il chip Agnus dovessero essere svolte dal 68000, questo risulterebbe 10 volte più lento.

Denise

Il chip Denise è un codificatore video che ha lo scopo di creare le videate che devono comparire sullo schermo di Amiga. Inoltre, Denise ha il compito di gestire gli sprite e il mouse.

Paula

Il chip Paula, a volte chiamato anche chip sonoro, ha il compito di gestire i quattro canali audio di Amiga con il supporto del chip Agnus (per i relativi quattro canali DMA dell'audio). Paula, inoltre, gestisce lo scambio dati con i disk driver e la porta seriale.

Nonostante i suoi tanti pregi anche Amiga ha però un "tallone d'achille": questi tre chip possono accedere solo RAM presente sulla scheda madre, ossia ai 512 Kb (il nome di CHIP RAM deriva appunto dal fatto che i chip custom possono indirizzare solo questi banchi di memoria). Questo, se vogliamo, è una delle maggiori limitazioni di Amiga: i dati di una musica o di una immagine devono risiedere per forza in questa memoria; è per questo che, nonostante si abbiano 3 Mb di RAM compare il messaggio "Not enought memory" quando tentiamo di caricare un'immagine HAM o di aprire delle finestre nuove.
Fortunatamente i nuovi modelli di Amiga hanno risolto questi problemi utilizzando nuove versioni di chip custom che sono in grado di indirizzare fino a 2 Mb di CHIP RAM.

Interfacciamento

Amiga è un computer che si può interfacciare molto facilmente con il mondo esterno. A seconda del modello troviamo comunque un'evoluzione verso la standardizzazione con il mondo degli altri computer. Vediamo ora modello per modello quali sono le possibilità di comunicazione di Amiga.

Amiga 1000

Come in tutte le macchine disponibili sul mercato, anche su Amiga (non solo 1000) le principali porte di connessione sono disposte sul retro dello chassis. Sulla destra della macchina troviamo le porte di connessione della tastiera e le porte del mouse e del joystick ove, oltre a questi dispositivi possono essere collegati penne ottiche o trackball; sul mercato però esistono anche digitalizzatori audio che utilizzano questa porta o programmi che forniscono un connettore che entra in questa porta come protezione contro la copia.
Sul retro troviamo la porta dei drive alla quale è possibile collegare altri tre drive (df1:, df2: e df3:) sia da 3"1/2 che da 5"1/4 dei quali però uno solo può sfruttare l'alimentazione di Amiga, mentre gli altri due devono avere una alimentazione propria. Altre due uscite sono quelle audio.
La porta parallela segue (o quasi) lo standard Centronics adottato dai PC

IBM a parte il fatto che anziché possedere un connettore femmina ne possiede uno maschio sui cui piedini sono presenti tensioni pericolose per una stampante.
Il connettore seriale, invece, segue lo standard RS-232 anche se, come nel caso della porta parallela, alcuni segnali non sono standard.
Un altro connettore presente sul retro di Amiga 1000 è quello RGB che manda i segnali al monitor; in questo caso non esiste uno standard ma si può dire che i creatori di Amiga quando hanno progettato la porta RGB hanno pensato veramente a tutto: oltre ai segnali per pilotare un monitor RGB sono infatti presenti quelli per gestire il Genlock. Ma non finisce qui: a fianco del connettore RGB sono presenti un connettore con il segnale video modulato in RF e uno con il segnale video composito a colori.
Infine, l'ultimo connettore porta tutti i segnali del bus di sistema; a questo connettore si possono collegare espansioni di memoria, hard disk e altro. Dal momento che Amiga 1000 non possiede slot d'espansione, la Commodore propose agli sviluppatori di aderire a uno standard denominato Zorro per le schede di espansione. In pratica l'utente comprava un cabinet esterno da collegare ad Amiga sul quale erano presenti slot d'espansione con questo standard.

Amiga 500

Per quanto riguarda l'interfacciamento, Amiga 500 si differenzia nelle porte seriale e parallela che hanno aderito completamente allo standard di mercato; è sparito il connettore video RF e quello composito è divenuto monocromatico. Per le espansioni del sistema Amiga 500 possiede due connettori, uno identico a quello di Amiga 1000 anche se non è consentito il collegamento con le periferiche di Amiga 1000, mentre il secondo permette il collegamento con una piccola espansione di memoria di 512 Kb (che porta il sistema a 512 Kb di RAM Chip e 512 di RAM Fast) con la batteria tampone per l'orologio interno.

Amiga 2000

Questa macchina è stata progettata per un uso professionale, per cui è lecito aspettarsi una configurazione che permetta un'espansione del

sistema ad alti livelli. Come configurazione di base Amiga 2000 possiede un Mb di RAM (512 Kb di Chip e 512 di Fast) e la possibilità di installare due drive internamente, anche se non varia il numero massimo di drive installabili, che rimane di quattro.
La differenza principale con A1000 e A500 è che Amiga 2000 possiede degli slot d'espansione che permettono di ampliare il sistema. Sulla scheda madre troviamo quindi uno slot video (sul retro della macchina è presente una porta RGB) al quale è possibile collegare un genlock, due slot IBM a 16 bit, sue slot IBM a 8 bit e 5 slot aderenti allo standard Zorro. Si noti, però, che lo standard Zorro di Amiga 2000 non è assolutamente compatibile con quello che Commodore aveva proposto con A1000, per cui tutte le schede che andavano bene per A1000 possono essere tranquillamente cestinate.
Un altro slot presente sulla scheda madre è lo slot CPU che contiene tutti i segnali che si trovano sugli slot d'espansione di A1000 e A500; a questo slot possono essere collegate schede acceleratrici, non ultime quelle dotate di 68040 che accelerano le prestazioni della macchina di circa 30 volte. Sugli slot Zorro invece possono essere montate schede di espansione di memoria, controller per hard disk e altre schede di vario tipo che aggiungono potenzialità alla macchina.
Un capitolo a parte meritano gli slot IBM; i due slot a 16 bit si trovano sullo stesso asse di due slot Zorro: la scheda che si collega a uno di questi slot contiene un vero e proprio PC IBM compatibile, con tanto di CPU e coprocessore, ed è in grado di emulare perfettamente un compatibile XT o AT (a seconda del tipo di scheda) realizzando un *ponte* (queste schede vengono appunto chiamate ***bridge-board***) tra Amiga e il PC compatibile su scheda.

Amiga 3000

Con l'uscita di Amiga 3000 si è fatto un enorme passo avanti nel mondo Amiga: dal vecchio 68000 si è passati al più recente e veloce 68030 affiancato dal coprocessore 68882. Per quanto riguarda le interfacce sono state aggiunte: un'uscita video standard VGA (Video Graphics Array) a 31.5 KHz e, finalmente, un'uscita SCSI ove poter collegare altri hard disk o cartridge removibili.

Le periferiche

Mouse

Nell'utilizzo di tutti i giorni, il mouse permette di controllare la freccia di selezione del Workbench, il sistema operativo grafico di Amiga. Quando il mouse viene spostato su di una superficie, la pallina al suo interno fa ruotare tre rotelline il cui movimento, e quindi quello del mouse, viene trasformato in impulsi riconoscibili dal computer. I segnali provenienti dal mouse (o da un dispositivo che funzioni sullo stesso principio, come una trackball) vengono inviati direttamente all'interno di contatori a 8 bit posti nei chip custom; tali contatori vengono incrementati o decrementati a seconda del tipo di impulsi. È quindi compito del sistema operativo leggere il valore dei contatori per determinare in quale direzione e con quale velocità l'utente ha mosso il mouse.

Tastiera

La tastiera comunica i dati che riceve a uno dei due Chip 8520 CIA (Complex Interface Adaptor). Questi due chip che vedremo nel capitolo 2 sono chip che gestiscono l'I/O (input/output, ossia il flusso di dati di ingrasso e di uscita) di Amiga. La tastiera comunica a uno di questi due chip in modo seriale. L'ultima cosa da dire sulla tastiera di Amiga è che non è una semplice periferica come in tutti gli altri computer: la tastiera di Amiga ha all'interno delle ROM e delle RAM; all'accensione della macchina la tastiera effettua un'autodiagnosi e segnala la presenza di qualche anomalia tramite il led del Caps Lock.
Ecco i significati:

- un lampeggio: la ROM della tastiera è guasta;
- due lampeggi la RAM della tastiera è guasta;
- tre lampeggi il timer di controllo non funziona;
- quattro lampeggi c'è un corto circuito tra due file di tasti o in altri tasti speciali di controllo.

Drive

Amiga, come è già stato detto in precedenza, è in grado di gestire un massimo di quattro disk drive. I drive possono essere sia da 5"1/4 che da 3"1/2. Il controllo dei drive avviene tramite una combinazione di porte su un chip 8520 e i DMA. Due porte sul 8520 selezionano il drive da cui mandare o ricevere i dati gestendo, tramite una serie di segnali la rotazione del motore, i movimenti delle testine, la presenza del disco o la protezione dalla scrittura.

Il trasferimento dei dati da e per il drive è realizzato tramite un canale DMA.

Capitolo 2

HARDWARE

In questo capitolo verrà analizzato più approfonditamente il funzionamento del copper e del blitter

Copper

Come già accennato nel primo capitolo, una delle caratteristiche peculiari dell'hardware di Amiga è la presenza di un processore che affianca il 68000. Questo coprocesore, denominato anche Copper, è parte integrante del Chip AGNUS, di cui si è già parlato nel capitolo precedente. Il Copper non è quello che si dice un vero microprocessore; infatti esso possiede un set di istruzioni alquanto limitato, solo tre: MOVE, SKIP e WAIT. A questo punto è lecito domandarsi a chi è venuto in mente di definire coprocessore un chip che ha solo tre istruzioni.

La risposta a questo quesito che sorge spontaneo è molto semplice: il Copper svolge delle funzioni fondamentali che occuperebbero parecchio tempo macchina. Per fare un esempio, il Copper gestisce la posizione sullo schermo del fascio di elettroni che rinfresca ogni linea del display molte volte al secondo. In realtà esistono altri dispositivi all'interno della complessa struttura di Amiga che potrebbero sostituirsi al Copper, ma non sarebbero comunque in grado di compararne la velocità.

Per capire quanto sia importante il Copper, è necessario ricordare che la maggior parte delle funzioni di Amiga sono gestite da dispositivi speciali che lavorano sulla base di precise informazioni, come la posizione in cui bisogna visualizzare un'immagine e l'istante in cui iniziare a visualizzarla. Se l'immagine venisse visualizzata immediatamente, non appena sono pronti i dati da visualizzare, si creerebbe un effetto noto come effetto "neve".

Il Copper, invece, sincronizza queste operazioni di visualizzazione dell'immagine, facendo sì che l'aggiornamento dei dati da visualizzare avvenga quando è finito il refresh dello schermo. Di seguito vengono illustrate brevemente le istruzioni del Copper.

WAIT

Wait è un'istruzione che aspetta finché non viene raggiunta una specifica posizione sullo schermo (il Copper ha infatti accesso diretto al segnale di sincronizzazione mandato al monitor; grazie a questa particolarità il Copper in qualsiasi momento sa dove si trova il fascio di elettroni dello schermo).

Nella maggioranza dei casi WAIT viene utilizzato insieme al blitter (che verrà trattato più avanti in questo capitolo).

MOVE

L'istruzione MOVE, che segue spesso l'istruzione WAIT, indica semplicemente al Copper di spostare un valore di 16 bit da un posto all'altro; i dati spostati sono solitamente informazioni che l'hardware specializzato necessita di avere nei suoi registri di memoria prima di svolgere un'operazione.

In questo modo un colore potrebbe essere posto in un registro colore, o un indirizzo posto in un registro utilizzato per specificare un'area di memoria contenente le informazioni di un'immagine.

SKIP

Skip è un'istruzione che non viene utilizzata molto spesso e permette di saltare a un'istruzione successiva se il display ha oltrepassato un certo punto. La funzione principale di SKIP, che solitamente opera in "collaborazione" con WAIT, è di individuare un'area dello schermo che, se raggiunta dal fascio di elettroni, non permette di inserire ulteriori immagini.

Queste tre istruzioni potrebbero sembrare non molto importanti se analizzate di per sé. Ciò accade perché non si ha ben presente quanto complesso è l'hardware video di Amiga e, di conseguenza, quanto sarebbe più complicato il lavoro della CPU senza l'aiuto del Copper.

Il Blitter

Come il Copper, anche il Blitter è fisicamente contenuto all'interno del chip AGNUS ed è un altro componente fondamentale dell'hardware di Amiga; è grazie ad esso e alla sua straordinaria velocità che Amiga può eseguire determinate operazioni, come le animazioni, che sono a livello di una workstation grafica.

Il nome BLITTER significa "Block Image Transferrer" (trasferitore di blocchi di immagini); il suo compito è, come si è potuto intuire dal nome, quello di trasferire blocchi di dati da una locazione di memoria a un'altra, sempre all'interno della memoria Chip. Anche il Blitter, come altri componenti dell'hardware di Amiga, è stato progettato per alleviare il lavoro al processore centrale ed essendo un chip appositamente progettato per svolgere determinate operazioni, svolge queste operazioni ben dieci volte più velocemente del 68000.

Come è già stato detto, il compito specifico del Blitter è lo spostamento diretto dei dati all'interno del sistema grafico di Amiga, anche se in realtà può spostare anche programmi all'interno della memoria.

La grafica rimane comunque il campo in cui il Blitter trova la maggiore applicazione: basti pensare a quando si sposta una finestra del Workbench che non è altro che un massiccio spostamento di dati da un punto all'altro nella memoria; ecco brevemente cosa accade quando si sposta una finestra:

- l'area che copriva la finestra viene ricostruita prelevando i dati da un buffer di memoria (rimozione della finestra);
- i dati dell'area che coprirà la finestra vengono salvati in un buffer di memoria;
- i dati che rappresentano i contenuti della finestra vengono scritti sul video.

Sebbene queste operazioni possano sembrare semplici, non sono sicuramente brevi e se non fossero eseguite alla velocità, a cui viaggia il Blitter, sarebbe molto spiacevole lavorare con Amiga.

In operazioni come lo spostamento di aree di memoria il Blitter ha bisogno di sapere in quale punto della memoria si trovano i dati da spostare, quale sia il punto di inizio dove effettuare la copia e quando iniziare a copiare. Per esempio potrebbe venir chiesto di copiare una piccole immagine di 32 x 20 pixel da un'area di memoria che rappresenta un display per un totale di 320 pixel. Per eseguire questa operazione il Blitter deve sapere dove l'indirizzo al quale trovare l'inizio del rettangolo contenente l'immagine da spostare, rimuovere due word (una word corrisponde a 16 bit) che corrispondono ai 32 pixel e copiarle all'indirizzo di destinazione, il tutto per 20 volte, senza toccare quanto sta intorno all'immagine da spostare.

Sorgenti multiple e operazioni logiche

Il semplice movimento dei dati non è il limite delle operazioni che è in grado di eseguire il Blitter. Uno dei problemi della scrittura di dati su un display sorge quando l'immagine che si deve spostare non è un rettangolo oppure quando le immagini che devono essere poste sullo schermo devono essere caricate da sorgenti separate.

Il Blitter può svolgere queste operazioni senza troppa fatica, in quanto il chip AGNUS dedica al Blitter ben quattro canali di accesso diretto alla memoria (DMA). Il Blitter utilizza tre di questi canali per prelevare i dati contemporaneamente da tre differenti sorgenti, mentre il quarto canale viene impiegato per porre i risultati in memoria. Nel caso di operazioni a sorgente multipla il Blitter è in grado di eseguire diversi tipi di operazioni logiche (AND, OR, INVERT, OR ESCLUSIVO) che permettono ai pixel di una sorgente di dati di sovrascrivere altri pixel, di essere alternati agli altri, combinati e così via.

Una delle applicazioni di queste proprietà del Blitter è quella di spostare elementi non rettangolari sullo schermo. Poniamo il caso che si debba muovere un cerchio sullo schermo: in questo caso la prima sorgente preleva una copia "nera" del cerchio, la seconda il cerchio per combinarli con ciò che si trova sullo schermo (terza sorgente) in modo che ogni pixel che risulti all'interno della maschera nera viene scritto prelevandolo dalla seconda sorgente, mentre tutti i pixel che stanno all'esterno della maschera rimangono intatti (terza sorgente). Il risultato viene poi inviato al quarto canale del DMA facendo così apparire sullo schermo il disegno circolare senza intaccare i pixel che stanno intorno.

Spostamento di bit

Il Blitter, come abbiamo visto, è molto veloce a compiere operazioni complesse di spostamenti di immagini su sfondi in modo da non disturbare le parti dello sfondo non coperte dall'immagine. Per tale motivo viene anche utilizzato per spostare elementi con gli stessi effetti degli sprite. Per fare ciò il Blitter sfrutta un'altra importante peculiarità che lo caratterizza, lo spostamento di un dato di un certo numero di bit a destra o a sinistra, prima di porlo sul display. Senza tale capacità, il più piccolo movimento possibile di un elemento che si potrebbe ottenere sarebbe una word (16 bit).

Grazie a questa caratteristica di spostare i dati a destra o a sinistra, un elemento può essere mosso di pixel in pixel come se fosse completamente distaccato dallo sfondo su cui si muove, con un'azione armoniosa

come uno sprite, sebbene l'elemento faccia parte dello sfondo. Gli elementi di questo tipo somiglianti agli sprite, il cui movimento è gestito dal Blitter sono detti "bob" ed esistono routine particolari per gestirle e dotarli di caratteristiche particolari, come trasparenza, distinzioni delle collisioni, priorità sullo schermo e così via.

Linee e riempimenti

Le ultime cose da dire sul Blitter sono le sue capacità di riempimenti di aree e disegno di linee. Il riempimento di aree è una logica estensione delle sue capacità di lavorare con il contorno di un elemento e di riempire un'area di memoria con qualcosa. La scrittura di linee è un'altra potenzialità del Blitter: come se non bastasse le linee che si possono tracciare hanno disegni diversi.

Capitolo 3

GRAFICA

Bitplane

Come si è visto nei capitoli precedenti, un'immagine che appare sul video viene ripetutamente tracciata da un continuo fascio di elettroni. Affinché ciò accada, l'immagine deve essere memorizzata in qualche parte della memoria, dove, in effetti, esiste una descrizione di ogni pixel; questa è costituita da uno o più bit a seconda del massimo numero di colori che si possono visualizzare. Il sistema grafico di Amiga permette di avere schermi, indipendentemente dalla risoluzione, a 2, 4, 8, 16 e 32 colori; se il numero dei colori è due, basterà un solo bit per definire il colore del pixel (0 spento, 1 acceso), mentre se il numero massimo di colori visualizzabili è 8 allora saranno necessari 3 bit. Nella figura 3.1 è riportato schematicamente il modo in cui Amiga ottiene le informazioni relative alla visualizzazione dei colori.

Come si può notare in memoria esistono tanti "piani di colore", o ***bitplane***, quanti sono i bit necessari per visualizzare il numero massimo dei colori. Per conoscere il colore di un pixel, Amiga estrae da ogni bitplane i bit corrispondenti a quel pixel e li raggruppa al fine di identificare il colore; l'insieme di questi bit forma un numero che assume il valore massimo di 32 (2^5) che identifica uno dei 32 (guarda caso) ***registri colore***. Questi ultimi, e tutti i registri in genere, non sono

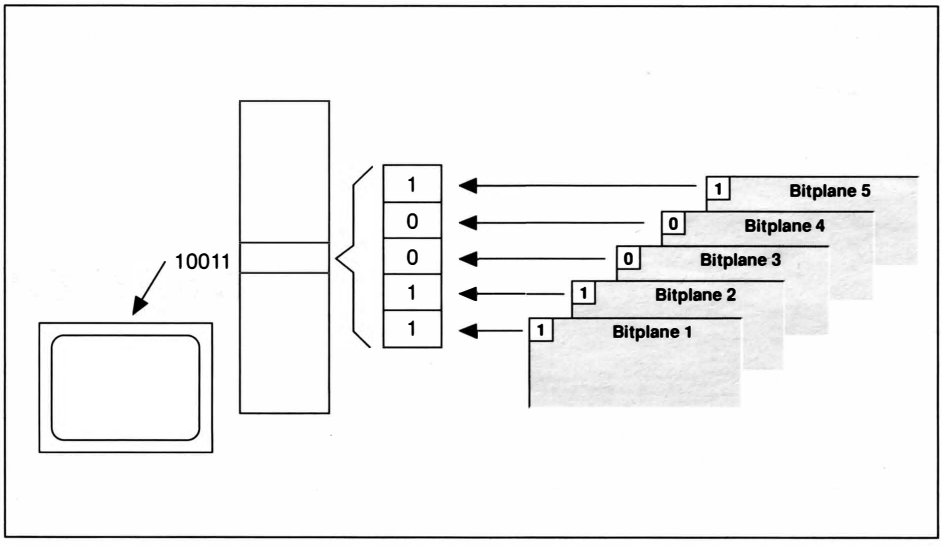

Figura 3.1
Visualizzazione dei colori

nient'altro che delle particolari locazioni di memoria all'interno dei circuiti integrati (non fanno parte né della memoria CHIP né di quella FAST) che contengono informazioni; i registri contengono, solitamente, 8, 16 o 32 bit, anche se non tutti sono necessariamente utilizzati.
Nella figura 3.2 si può osservare come sia composto schematicamente un registro colore e quale significato abbiano i bit che vi troviamo all'interno.

Come si vede i registri colore sono composti da 12 bit, divisi idealmente in tre gruppi di quattro bit; ciascuno di questi gruppi rappresenta l'intensità di blu (bit da 0 a 3), di verde (da 4 a 7) e di rosso (da 8 a 11). Ogni colore della terna RGB (Red, Green, Blue - rosso, verde e blu) può avere un massimo di 16 intensità di colore (2^4=16), che combinate danno origine a 4096 diversi colori che rappresenta il numero massimo di colori che il sistema grafico di Amiga può visualizzare.

Un esempio pratico

Si supponga di voler visualizzare due pixel, uno di colore azzurro e l'altro di colore rosso vivo. Innanzitutto si devono specificare le componenti cromatiche dei due colori: per l'azzurro (che si ottiene "mescolando"

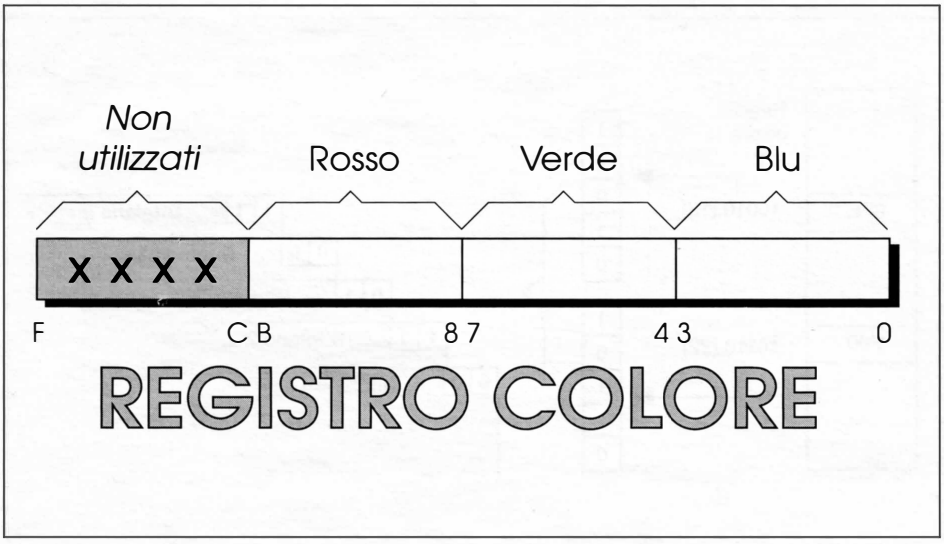

Figura 3.2
Struttura dei registri colore

varie componenti cromatiche) si dovrà impostare il rosso con intensità 6, il blu con intensità 14 e il verde con intensità massima, 15. Per quanto illustrato nella figura 3.2, il codice colore dell'azzurro è 6FE. Il rosso, in quanto componente cromatica, si ottiene impostando al massimo questa componente cromatica e al minimo le altre due, verde e blu: il codice colore risultante sarà pertanto F00.

A questo punto questi codici devono essere memorizzati nei registri colori; se lo schermo è a 32 colori (5 bitplane) i due colori scelti possono essere memorizzati in uno qualunque dei 32 registri (tenendo presente che, comunque, il registro 0 imposta il colore di sfondo). Nel caso in cui lo schermo fosse a 16, 8, 4 o due colori, i codici ottenuti verranno memorizzati, rispettivamente, nei primi 16, 8, 4 o 2 registri colore. Poniamo nel caso che stiamo analizzando, che lo schermo in questione sia a 5 bitplane e che i codici colore dell'azzurro e del rosso vengano memorizzati, rispettivamente, nei registri 18 e 22, come mostra la figura 3.3.

Il valore 18, quindi, che in binario è 10010, dovrà essere scritto nei cinque bit associati al primo pixel che sono i primi bit di ogni bitplane; dal momento che i bitplane con numero d'ordine più alto contengono i bit

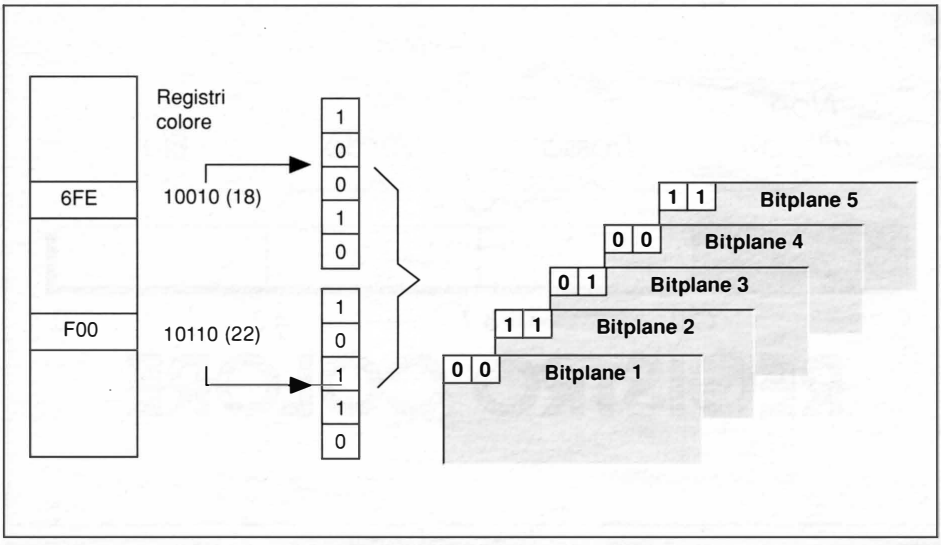

Figura 3.3
Esempio di visualizzazione dei colori

più significativi, si avrà:

- primo bit del quinto bitplane uguale a uno
- primo bit del quarto bitplane uguale a zero
- primo bit del terzo bitplane uguale a zero
- primo bit del secondo bitplane uguale a uno
- primo bit del primo bitplane uguale a zero

Lo stesso discorso vale per il secondo pixel dello schermo, che come si è detto è di colore rosso memorizzato nel registro colore 22 (10110 in binario); in conclusione si avrà:

- secondo bit del quinto bitplane uguale a uno
- secondo bit del quarto bitplane uguale a zero
- secondo bit del terzo bitplane uguale a uno
- secondo bit del secondo bitplane uguale a uno
- secondo bit del primo bitplane uguale a zero

Il tutto è rappresentato schematicamente in figura 3.3.

Creazione di uno schermo

Prima di generare uno schermo, bisogna decidere se questo dovrà essere in bassa o alta risoluzione: se lo schermo dovesse essere in bassa risoluzione avrebbe 320 pixel sulle righe orizzontali, contro le 640 dell'alta risoluzione; in quest'ultimo caso il massimo numero di bitplane scende a 4 e di conseguenza il massimo numero di colori visualizzabili diventa 16.

Per quanto riguarda la risoluzione verticale, questa può essere scelta tra i 256 pixel e 512; si noti che negli Statu Uniti, a causa del diverso standard televisivo vigente (NTSC contro il PAL europeo) la risoluzione verticale è di 200 o 400 linee. Ciò si nota soprattutto con i primi programmi d'oltreoceano che non sfruttano la parte bassa dello schermo.

Come si diceva, la risoluzione verticale che si ha a disposizione è di 256 o 512 linee; la prima risoluzione è quella standard, mentre la seconda si ottiene mediante l'interlacciamento dello schermo. Questa tecnica consiste nel visualizzare alternativamente le linee pari e quelle dispari. Questo però produce sui monitor ad alta risoluzione un fastidioso sfarfallio (filckering) che si accentua maggiormente nei contrasti di colore; per ovviare a tale inconveniente esistono due modi: il primo consiste nell'installare un apposita scheda (solo all'interno di A2000 e A500, in quanto A3000 ne possiede già uno internamente) detta "flicker fixer", mentre la seconda soluzione può essere quella di utilizzare un monitor ad alta persistenza, che, però, oltre a essere più costoso produce un effetto detto "stella cadente" (tutto ciò che si muove sullo schermo lascia la scia...).

A questo punto non resta che scegliere il numero dei bitplane di cui dovrà essere costituito lo schermo e di conseguenza il massimo numero dei colori.

Il rapporto tra il numero dei bitplane e quello dei colori è fornito dalla seguente relazione:

$$\text{n.colori} = 2^{\text{n-bitplane}}$$

$$\text{n-bitplane} = \log_2 \text{n.colori}$$

Dopo aver deciso il numero dei bitplane, Amiga scriverà questo numero in un apposito registro e verranno creati i bitplane in memoria. Un bitplane non è nient'altro che un'area di memoria che contiene i dati relativi al contenuto dello schermo. A differenza di altri calcolatori, Amiga non ha un'area fissa per memorizzare questi dati, ma l'indirizzo di partenza viene deciso a seconda della disponibilità di memoria della macchina; l'unica cosa certa è che i bitplane sono memorizzati nella CHIP RAM (per i motivi che sono stati ampiamente descritti nei capitoli precedenti). Può accadere che non ci sia abbastanza memoria libera per aprire uno schermo e allora il sistema risponderà con un "Not enough memory", oppure come nel caso di Deluxe Paint, verrà aperto uno schermo con un numero di bitplane inferiore. L'indirizzo iniziale di ogni bitplane viene memoriizzato in appositi registri.

Come appena detto, l'indirizzo iniziale dei bitplane non è mai lo stesso, per cui Amiga tiene traccia dei bitplane grazie a sei puntatori appositamente creati per indicare l'indirizzo, in memoria, dei bitplane. Nei tipi di schermi fino a ora descritti sono utilizzati un massimo di cinque bitplane; il sesto bitplane viene utilizzato sono in modalità grafiche particolari. Questi puntatori contengono inizialmente l'indirizzo del primo bit di ogni bitplane. Dopo che è stato letto il primo bit da ogni bitplane (per formare il numero del registro che contiene il colore del primo pixel) tutti i puntatori ai bitplane vengono incrementati in modo da puntare al bit successivo, per determinare il colore del secondo pixel. Una volta che si arriva all'ultimo pixel il puntatore viene reinizializzato all'indirizzo iniziale durante l'intervallo di ***vertical blank*** (l'intervallo di tempo in cui il raggio di elettroni viene spostato dal fondo all'inizio dello schermo).

La modalità HAM

Fino a ora si è visto che il numero massimo di colori visualizzabili sullo schermo è trentadue. Esiste però una modalità grafica particolare grazie alla quale è possibile visualizzare tutti i 4096 colori che si hanno a disposizione su Amiga: tale modalità è denominata ***Hold And Modify*** (HAM).

Normalmente il colore di un pixel viene determinato esclusivamente dalle informazioni contenute nei bitplane e di conseguenza dalle impostazioni dei registri colore. Nella modalità HAM, invece, i pixel, oltre ad assumere i colori nella maniera tradizionale, possono visualizzare i colori in base al colore del pixel che lo precede secondo le modalità indicate dai bitplane.

In questa modalità vengono utilizzati tutti e sei i bitplane in bassa risoluzione. I bit estratti dai bitplane 6 e 5 indicano il modo in cui devono essere interpretati i restanti bit dei quattro bitplane. Se questi bit sono 00 allora i quattro bit dei restanti bitplane verranno considerati nella consueta maniera: la combinazione di questi bit formerà un numero da 0 a 15 che indicherà il numero del registro dal quale prelevare il colore. Se invece i bit dei bitplane 6 e 5 è 01, il colore del pixel corrisponderà a quello precedente ma con la componente del blu sostituita dal valore indicato dai quattro bit dei bitplane rimanenti.
Come si è visto, il colore di un pixel è determinato da una terna di gruppi di quattro bit rappresentante, ognuno, l'intensità di rosso, verde e blu. Nel caso che si è appena visto, il pixel in questione avrà le componenti di rosso e di verde identiche a quelle del pixel precedente, mentre la componente di blu nel registro colore (bit 0-3) sarà indicata dai bit dai quattro bitplane che avanzano.

Lo stesso discorso vale per le componenti di rosso e di verde. Se i bit dei bitplane 6 e 5 indicano 10 allora la componente di colore interessata alla variazione sarà il rosso (bit 8-11 dei registri colore, si veda la figura 3.2), mentre se questi bit indicano 11 la componente di colore che varierà sarà il verde (bit 4-7). Il tutto viene spiegato più schematicamente nella tabella 3.1.
Grazie a questa modalità è possibile visualizzare contemporaneamente sullo schermo tutti i 4096 colori.

Come si è visto, il sistema grafico di Amiga è molto complesso; la cosa più importante che è doveroso specificare è che tutto quello che di cui si è trattato in questo breve capitolo non è che solo la punta di un iceberg.

Bitplane 6 e 5	Bitplane 4 3 2 1	Descrizione
00	XXXX	il pixel assume il colore del registro colore XXXX
01	XXXX	il pixel assume il colore del pixel precedente con intensità del blu uguale a XXXX
10	XXXX	il pixel assume il colore del pixel precedente con intensità del rosso uguale a XXXX
11	XXXX	il pixel assume il colore del pixel precedente con intensità del verde uguale a XXXX

Tabella 3.1
Significato dei bit nella modalità HAM

In realtà non solo tutto il sistema grafico è molto più ampio di quello che questa introduzione alla grafica ha voluto mostrare, ma il tutto è molto più complesso.

Capitolo 4

AUDIO

Le capacità sonore di Amiga sono notevolmente superiori alla media degli altri personal computer; la motivazione principale risiede nel fatto che Amiga genera il suono in modo digitale, come fa Macintosh, e non analogico, come il resto dei personal computer. Per comprendere come Amiga produce un suono conviene fare un passo indietro.

Suono analogico e digitale

I suoni che percepiamo sono prodotti dalla variazione della pressione dell'aria. L'onda sonora che rappresenta questa variazione continua è riportata in figura 4.1. Una tipica forma d'onda sonora è sinosuidale ed è quella che verrà utilizzata per gli esempi di questo capitolo. Nel grafico di figura 4.1 l'asse delle ascisse indica il tempo e quello delle ordinate la variazione dell'ampiezza della forma d'onda.

Il problema che si pone a questo punto è questo: come può Amiga rappresentare una variazione continua di tensione (poiché i suoni vengono trasformati in tensione dai microfoni), quindi una grandezza analogica, se lavora in modo digitale, ossia solo con 0 e 1?

La risposta a questo quesito è abbastanza semplice. Dalla forma analogica viene letto il valore di tensione a intervalli di tempo molto piccoli; alla fine si otterrà una tabella con una serie di numeri (grandezza digitale).

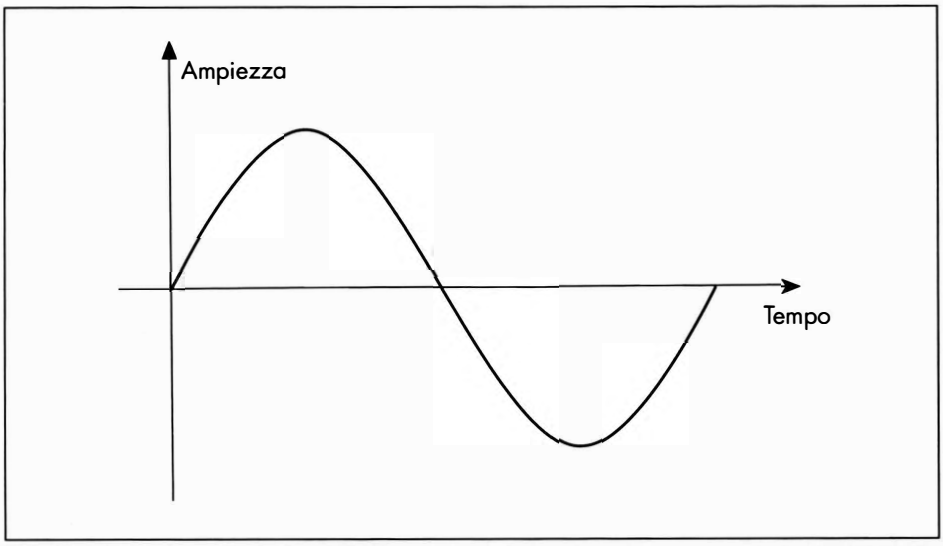

Figura 4.1
Una forma d'onda sonora

Questo processo di ***digitalizzazione*** (ossia di conversione da una grandezza analogica a una digitale) viene chiamato ***campionamento*** e viene effettuato da appositi circuiti chiamati ***ADC*** (Analog/Digital Converter) o convertitori A/D.

Se poi si inviano i valori appena ottenuti con gli stessi intervalli di tempo a un convertitore D/A, ***DAC*** (Digital/Analog Converter) si ottiene una forma d'onda simile a quella rappresentata in figura 4.2.

Lo scopo del processo di campionamento è, quindi, quello di ottenere un valore costituito da un certo numero di bit che rappresenti il valore di tensione in un istante specifico della forma d'onda analogica. È chiaro che più è elevato il numero di bit che rappresenta il valore di tensione più ampia sarà la gamma dinamica (differenza tra il valore più alto e quello più basso) campionabile. Un altro parametro da tenere in considerazione in questa fase è la ***frequenza di campionamento***, ossia l'intervallo di tempo tra un campionamento e l'altro: più è elevata la frequenza (ossia più breve è l'intervallo) e maggiore sarà la fedeltà del suono campionato.

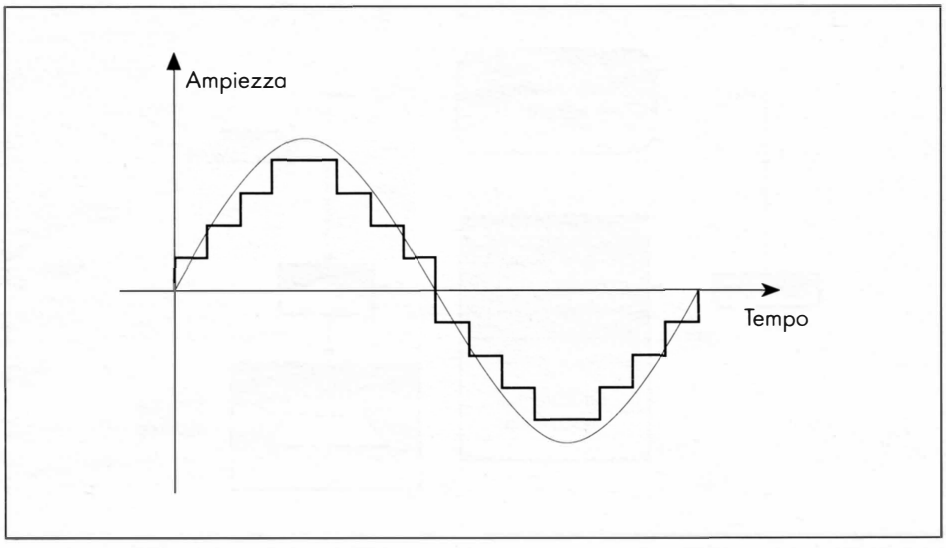

Figura 4.2
Forma d'onda ottenuta dopo il campionamento

Il compact disc, che funziona in modo digitale (il laser che legge il disco restituisce 0 o 1), possiede dei convertitori D/A a 16 bit, permettendo così di ottenere 65536 valori diversi, e utilizza una frequenza di campionamento di poco superiore ai 44 KHz (in un secondo vi sono più di 44000 letture dal disco).

L'audio di Amiga

Amiga, nel suo piccolo, possiede quattro convertitori D/A a 8 bit (256 valori diversi) e ha una frequenza massima di campionamento pari a 28,867 KHz. Questi quattro convertitori sono programmabili indipendentemente l'uno dall'altro e possiedono, ciascuno, un canale audio di uscita ben definito. I canali audio, numerati da 0 a 3, sono accoppiati per ottenere un'uscita stereo: i canali 1 e 2 formano l'uscita di destra, mentre i canali 0 e 3 l'uscita di sinistra. Come è stato accennato nei primi capitoli, i quattro convertitori si trovano fisicamente all'interno di Paula e a ognuno di essi è associato un canale DMA.

Esistono due metodi per la creazione di un suono: il primo, e il più utilizzato, è detto *automatico* e il secondo è detto *manuale*.

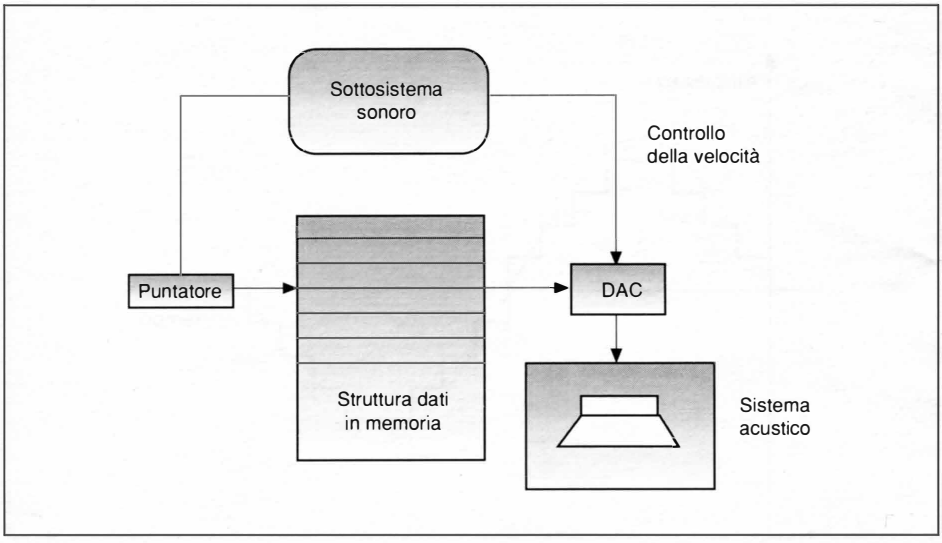

Figura 4.3
Schema di funzionamento per la creazione di un suono

Creazione di un suono

Metodo automatico

La creazione di un suono con il metodo automatico prevede alcuni passi ben definiti:

- decidere quale canale DMA utilizzare (si ricordi che questa scelta deve essere fatta in base al canale stereo sul quale di intende udire il suono);
- definire la forma d'onda sonora e creare conseguentemente in memoria un'apposita struttura dati;
- informare il sistema dove poter reperire questi dati;
- impostare alcuni registri specifici che comunichino a un sottosistema sonoro dove reperire i dati (il sottosistema sonoro ha il compito di gestire alcuni parametri sonori, come la velocità di lettura dei dati, che modificano notevolmente il suono finale);
- scegliere il volume;
- impostare la frequenza di campionamento;
- selezionare il canale audio e far partire il DMA.

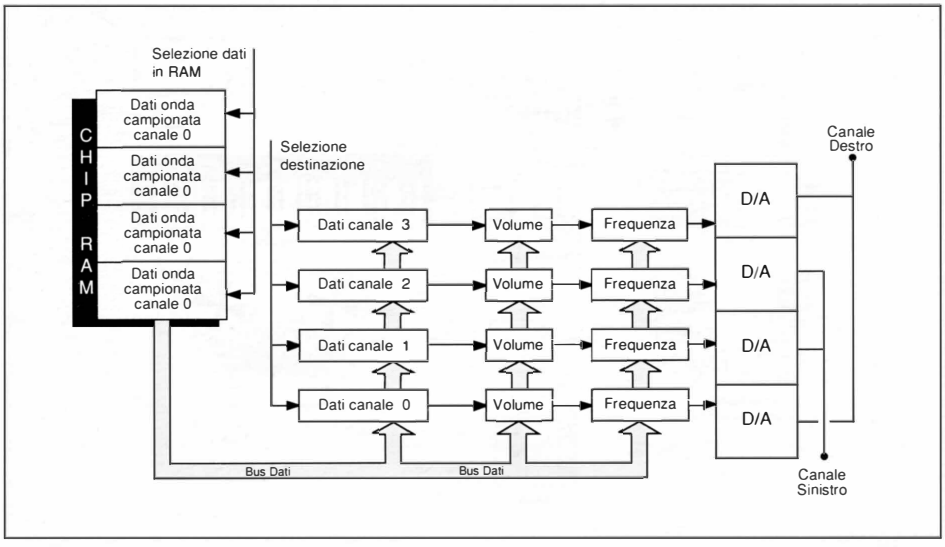

Figura 4.4
Circuiteria interna dei canali DMA

Il sistema sonoro può quindi essere visto come un puntatore in memoria (gestito dal sottosistema sonoro) che preleva i valori a una velocità ben definita (impostata sempre dal sottosistema sonoro) e li spedisce al convertitore D/A per convertire in tensione i valori numerici, affinché possano essere inviati agli altoparlanti. Naturalmente, visto che i quattro canali audio sono programmabili in modo indipendente l'uno dall'altro, è possibile generare quattro diversi segnali audio contemporaneamente.

È importante specificare che il metodo automatico di creazione di un suono prevede la descrizione di un ciclo completo della forma d'onda. Fatto questo passo il suono viene generato senza altri interventi esterni da parte dell'utente.

Nella figura 4.3 viene riportato lo schema di funzionamento per la creazione di un suono, mentre nella figura 4.4 è rappresentata schematicamente la circuiteria interna relativa ai canali audio.

Metodo manuale

A differenza del metodo automatico, quello manuale prevede la descri-

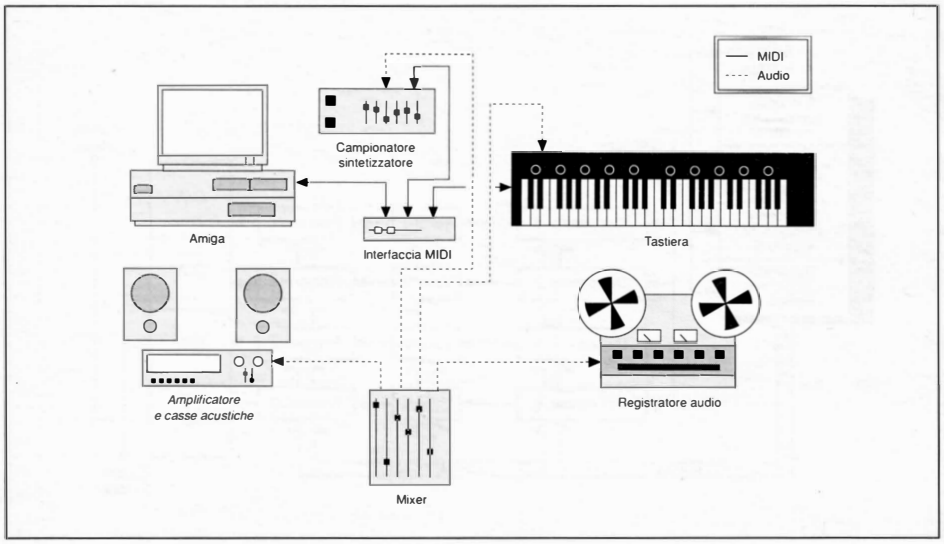

Figura 4.5
Sistema Amiga-MIDI

zione istante per istante dell'onda sonora da generare. L'utente deve quindi scrivere direttamente nei convertitori D/A i valori calcolati in base alla tabella in memoria. In questo modo si possono ottenere dei risultati sonori impensabili; il grande difetto però è che questo metodo sottrae molto "CPU time", ossia impegna molto la CPU (in quanto non vengono utilizzati i canali DMA ma la CPU per scrivere i dati nei convertitori) per cui viene scarsamente utilizzato.

Il MIDI

L'ultima sofisticazione tecnologica nel campo musicale è il MIDI (Musical Instrument Digital Interface), un linguaggio e un insieme di protocolli che permettono a dispositivi musicali MIDI compatibili di comunicare tra di loro. Amiga può facilmente diventare il centro di un sistema musicale MIDI collegando un'interfaccia MIDI alla porta seriale e i dispositivi MIDI all'interfaccia, come mostrato in figura 4.5. Una volta che Amiga è stato "MIDIzzato", è possibile suonare su di uno strumento MIDI e vedere le note apparire sullo schermo di Amiga, utilizzando, ovviamente, il software adatto.

Un'altra possibilità è quella di creare dei suoni con Amiga e utilizzarli con una tastiera MIDI compatibile. Qualsiasi interfaccia MIDI offre molti canali MIDI e porte MIDI in/out per il collegamento di apparecchiature MIDI.
Dopo Amiga e l'interfaccia MIDI, il pezzo più importante in un sistema MIDI è una tastiera MIDI compatibile. Alcuni programmi permettono di utilizzare la tastiera di Amiga come una tastiera musicale, anche se ciò non è raccomandabile neanche per i principianti. Sul mercato si trovano tastiere MIDI compatibili anche a basso prezzo.

Le tastiere possono anche contenere il proprio campionatore e sintetizzatore. Recentemente sono stati resi "MIDI compatibili" anche equipaggiamenti tipo ottoni, percussioni, strumenti a corda e a fiato; ovviamente è possibile utilizzare uno qualsiasi degli strumenti MIDI, non solo le tastiere.

Software MIDI

Come è appena stato detto, per poter utilizzare un'interfaccia MIDI con i relativi strumenti è necessario utilizzare del software specifico. Questi si dividono per tipi a seconda delle funzioni che svolgono:

- *sequencer*, per il controllo del suono proveniente dai dispositivi MIDI;
- *sintetizzatori*, chiamati anche synth, per la creazione di suoni e musica;
- *drum synthesizer*, per effetti di percussione automatici;
- *note editor*, (notazione musicale) per comporre, modificare, trascrivere e suonare le partiture musicali;
- *patch editor/librarian*, per il collegamento a equipaggiamenti audio (un patch è la connessione tra due o più tracce o sorgenti audio). Un patch editor è specifico per lo strumento musicale collegato con esso.

La maggior parte del software musicale di Amiga è MIDI compatibile, come Sonix, per esempio. Se si ha intenzione di utilizzare professionalmente il MIDI come strumento di lavoro, allora è necessario disporre di

un sequencer MIDI. Tra questi prodotti si segnalano Bar & Pipes1.0 e la relativa versione *Professional*, KCS 3.5 e KCS Level II 3.5, Master Track Pro, MIDI Recording Studio, Music X, Pro 24 III, Pro MIDI Studio, Studio 24, Tiger Cub, Track 24 e Trax.

Per quanto riguarda i programmi di notazione musicale sono disponibili The Copyist III, Deluxe Music Construction Set 2.5 e Studio 24. Tra gli editor/librarian per sintetizzatori si segnalano, Super Librarian, Casio CZ Rider, Casio VZ Rider, Emu Proteus 1/XR, Esqapade, Lexicon PCM70, Kawai K1, Kawai K5, K1 Master Editor/librarian, Korg M1, Oberheim M-6 Tricks, Oberheim Matrix 6/1000 Master editor/librarian, Roland D-10/110/20, Roland D-10/110/20 Master editor/librarian, Roland D-50/550, Roland MT-32, Yamaha DX Heaven, Yamaha DX7II/TX802, 4-Op Deluxe. Il mercato degli editor/librarian per campionatori offre, invece, MIDI Sample Wrench, Synthia II e Synthia Professional.

Per tutti i riferimenti a questi prodotti si consiglia di consultare le riviste del settore di informatica musicale.

L'ultima cosa da precisare è che, come tutti gli altri settori, anche quello dell'informatica musicale è in continuo sviluppo; probabilmente quando saranno lette queste righe vi saranno una decina di prodotti nuovi o, comunque, aggiornati.

Capitolo 5

L'EXEC

In questo capitolo viene trattato un argomento importantissimo: l'Exec, il cuore del sistema operativo di Amiga. Ne segue, purtroppo, che non tutti gli argomenti trattati, per la loro implicita complessità, saranno spiegati in modo semplice come è accaduto per gli argomenti discussi nei capitolo precedenti: tuttavia, sarà fatto tutto il possibile affinché gli argomenti trattati d'ora in avanti siano il più chiaro e semplice possibile.

Una breve introduzione

Per spiegare in maniera appropriata cos'è e quale funzione ha l'Exec è necessario fornire prima una vista d'insieme della struttura gerarchica del software e dell'hardware di Amiga; tutto ciò è rappresentato nel diagramma a blocchi di figura 5.1. Si può notare che i vari livelli del software sono disposti l'uno sull'altro; in cima vi sono i programmi applicativi che comunicano con l'utente, mentre all'ultimo livello (chiamato livello primario) si trova l'hardware della macchina.

Il livello primario

L'interfacciamento con l'hardware del sistema di Amiga avviene tramite alcune routine specifiche. L'insieme delle routine che controllano e gestiscono l'utilizzo del 68000 è chiamato Exec; la sua funzione è quella di ripartire il 68000 tra i vari programmi (task) che si trovano in funzione contemporaneamente: grazie all'Exec, dunque, Amiga è in grado di

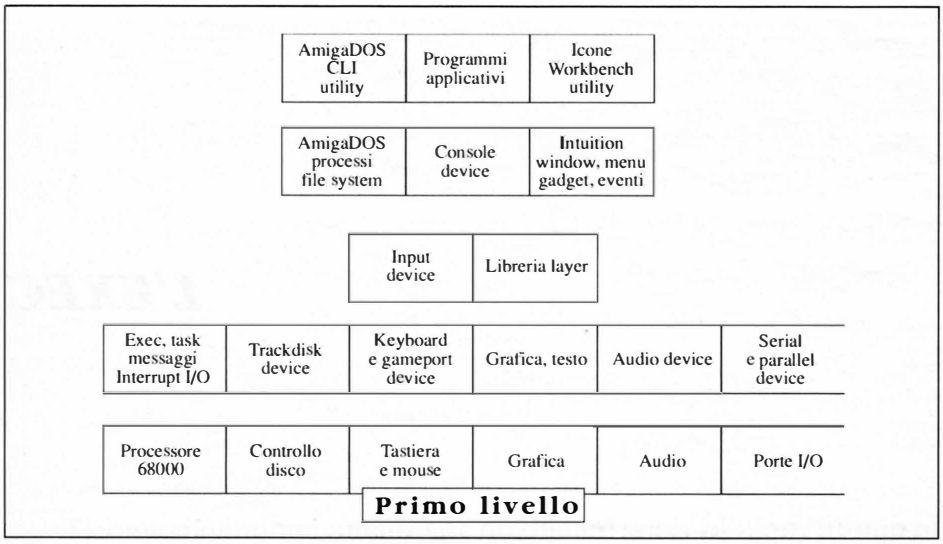

Figura 5.1
Gerarchia del sistema software

operare in multitasking, caratteristica che tuttora è invidiata dai sistemi operativi concorrenti. Inoltre l'Exec gestisce anche l'allocazione della memoria per ogni singolo programma, controlla le varie richieste di interrupt che vengono generate dall'hardware o dalle applicazioni software, mantiene le liste dei programmi che sono in attesa di determinati eventi, la lista delle aree di memoria disponibili, le liste dei messaggi relativi a ogni applicazione e le liste degli input, come gli spostamenti del mouse o la pressione dei tasti.

Allo stesso livello dell'Exec si trovano i device, unità software che gestiscono i drive, la tastiera, il mouse e le porte giochi, le uscite audio, la porta seriale e quella parallela. Per finire si trovano le routine di gestione della grafica (tracciamento dei disegni, visualizzazione delle aree grafiche, selezione dei colori, manipolazione degli oggetti grafici, ecc.).

Gli altri livelli

Nel secondo livello sono collocati i device di input e quelle funzioni dette di layer. I device di input girano come programmi indipendenti e provvedono a convogliare tutte le informazioni di input in un unico canale. Le informazioni di input arrivano da Intuition o da un device di console.

Le funzioni di layer si trovano in cima al sistema grafico e gestiscono quelle routine che dividono le normali aree grafiche in zone multiple, riunendo quelle aree vengono utilizzate per la costruzione delle finestre. Tra le funzioni di layer si segnalano quelle per la creazione, la gestione e lo scambio fronte-retro delle aree.

Al terzo livello si trova l'AmigaDOS, Intuition e i device di console. AmigaDOS è un sistema operativo in grado di operare in multitasking; scambiando varie informazioni con l'Exec è in grado di ripartire in modo ottimale tutte le risorse e le richieste del sistema, non ultima la ripartizione del lavoro del microprocessore. L'AmigaDOS, inoltre, si occupa della gestione dei file e dell'esecuzione dei programmi.
Intuition rappresenta l'interfaccia multi-schermo e multi-finestra di Amiga; è costituito da un'insieme di funzioni (libreria) tramite le quali è possibile creare un'interfaccia grafica.

I device di input hanno la funzione di convogliare tutte le informazioni di input a Intuition che, a sua volta, "filtra" le varie informazioni al singolo programma. Il device di console, invece, è un particolare device di input che si può applicare a una finestra in modo che essa funzioni come un terminale: digitando da tastiera i caratteri appaiono a video (l'esempio più evidente è il CLI). In questi casi il compito di Intuition è quello di tenere attiva una solo finestra alla volta in modo che quanto digitato venga spedito a uno specifico device di console.

Nel livello massimo, per finire, si possono trovare i programmi applicativi, il Workbench e il CLI (Command Line Interface).
Il CLI è un'interfaccia a linea di comando, ossia la tradizionale interfaccia dei computer, quella che ricorda i terminali. Esso è in realtà un'applicazione che ha il compito di tradurre ciò che viene inserito da tastiera in comandi che il computer possa eseguire. Il Workbench sostituisce con un'interfaccia a finestre e icone tutte le operazioni che si svolgono tradizionalmente con l'interfaccia a linea di comando.
Un programma applicativo presenta il tipo di interfaccia deciso dal programmatore ma basata dal tipo di programma stesso.

L'EXEC

L'Exec è una struttura multitasking basata su liste. Tutto ciò che viene gestito dall'Exec si trova su una lista.
Il multitasking, invece, è la possibilità di far girare due o più programmi (detti task), eseguendoli singolarmente per un brevissimo intervallo di tempo, in modo da farli apparire come funzionanti contemporaneamente.

Le principali funzioni dell'Exec sono:

- allocare la memoria su richiesta dei vari task che stanno girando, in base a una lista dei blocchi di memoria liberi;
- gestione dei task attraverso la lista dei task che stanno girando, che sono pronti per essere eseguiti o che stanno attendendo un segnale prima di poter di nuovo partire;
- controllo della lista di librerie di funzioni dalle quali i vari task utilizzano in comune diversi codici di programma;
- gestione della lista dei device (device driver) che i task devono utilizzare per l'Input/Output (I/O) di sistema;
- controllo della lista degli Interrupt per gestire i vari interrupt hardware e software generati dal 68000, dai chip custom e dal software di sistema.

Come è stato detto, tutte le strutture e le operazioni gestiste dall'Exec sono basate sulle liste; lo scopo di questa scelta, da parte dei programmatori del sistema operativo di Amiga, è quello di avere una configurazione dinamica di tutte le strutture, in modo da non avere limitazioni arbitrarie.

Le liste

Le liste sono composte da due tipi di strutture dati: l'intestazione e i nodi, una catena collegata in due direzioni. L'intestazione contiene il puntatore al primo e all'ultimo nodo.

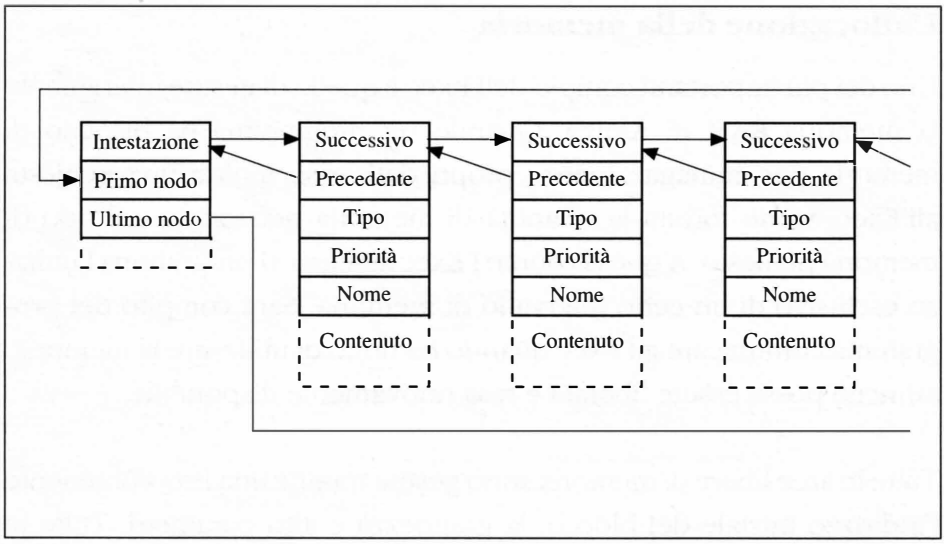

Figura 5.2
Esempio di lista

Ciascun nodo, che può essere posizionato in qualsiasi zona della memoria, è diviso in tre parti: collegamenti, informazioni e contenuto. La parte dei collegamenti contiene due puntatori: uno che individua il nodo successivo e l'altro che punta il nodo precedente; subito dopo questi due indirizzi sono contenute le informazioni sul nodo: tipo del nodo, priorità e puntatore al nome del nodo. L'ultima parte contiene i veri e propri dati del nodo. La figura 5.2 mostra schematicamente la composizione di una lista.

Grazie alle liste è quindi possibile inserire dinamicamente nuovi anelli alla catena: se, per esempio, si vuole inserire un nuovo elemento (nodo) tra il terzo e il quarto elemento, sarà semplicemente necessario creare una nuova struttura "nodo" in memoria e aggiornare il "puntatore al nodo successivo" del terzo nodo e il "puntatore al nodo precedente" del quarto nodo con l'indirizzo del nuovo nodo.

Anche se può sembrare macchinoso e complesso, questo tipo gestione di tutte le strutture dati permette di non avere limitazioni fisse delle strutture stesse (se non per esaurimento della memoria).

L'allocazione della memoria

Uno dei più importanti compiti dell'Exec è quello di gestire interamente la memoria RAM di Amiga. Quando un programma ha bisogno di memoria per immagazzinare i propri dati, esso inoltra una richiesta all'Exec: viene fornita la quantità di memoria necessaria e il tipo di memoria richiesto. A questo punto l'Exec assegna al programma l'utilizzo esclusivo di un certo intervallo di memoria. Sarà compito del programma comunicare all'Exec quando ha finito di utilizzare la memoria, affinché possa essere liberata e resa nuovamente disponibile.

Tutte le aree libere di memoria sono gestite tramite una lista contenente l'indirizzo iniziale del blocco, la grandezza e altri parametri. Tutte le funzioni di allocazione della memoria restituiscono al programma richiedente, l'indirizzo di inizio del blocco appena assegnatogli.
Come è stato detto, è necessario restituire la memoria richiesta non appena se ne è finito l'uso. Questa operazione è importante perché l'Exec non tiene traccia di quale task ha della memoria allocata e quindi non può sapere quale task non restituisce la memoria. Se ciò avviene, ossia un'applicazione non rilascia la memoria che aveva in gestione, la memoria non restituita rimarrà inutilizzabile finché non verrà resettata la macchina.

Le librerie

Il sistema software di Amiga è suddiviso in diversi sottosistemi conosciuti come librerie e device. Una libreria è un gruppo di funzioni che trattano dello stesso argomento che possono essere utilizzate contemporaneamente da più applicazioni. Le librerie sono composte dalle routine (codice rientrante, ossia eseguibile da più task contemporaneamente) e dalla base della libreria; quest'ultima contiene una struttura nodo di libreria, la quale è preceduta da una tabella di vettori di 6 byte relativi alle funzioni della libreria e seguita da specifiche strutture di libreria.

Alcune librerie risiedono in memoria, mentre altre vengono caricate da disco in memoria all'occorrenza, per essere successivamente rimosse se nessun altro task le sta utilizzando.

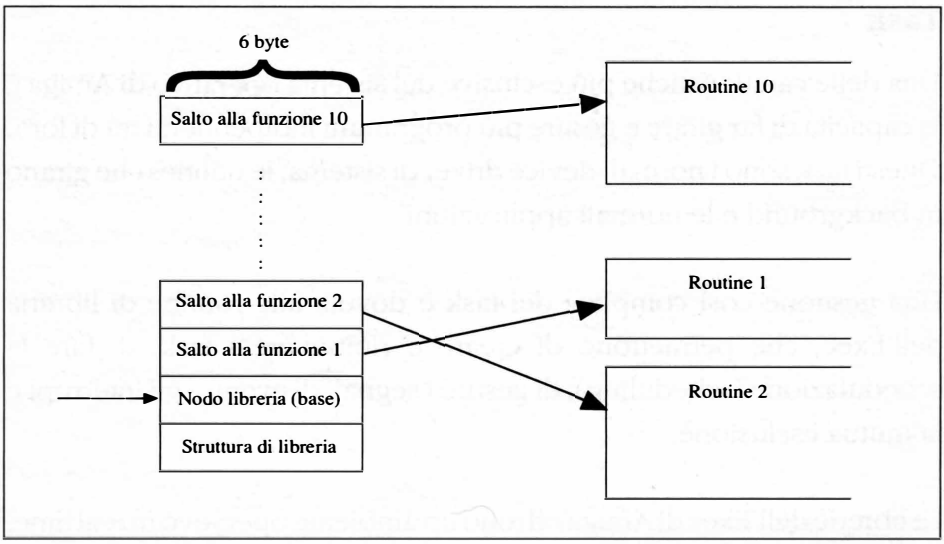

Figura 5.3
Schema di una libreria

Quando un programma ha bisogno di una routine contenuta in una libreria, esso utilizza a sua volta una routine della libreria dell'Exec (exec.library) automaticamente aperta del sistema all'avvio della macchina; questa routine restituisce al programma l'indirizzo di base della libreria.

A questo punto il programmatore (o il compilatore se non si sta utilizzando l'assembler) sa che la routine di cui si ha bisogno è, per esempio, la decima della libreria. Per accedere a questa routine il sistema moltiplicherà il numero della routine (10) per 6 (dimensione in byte dell'elemento della jump table) ottenendo sessanta che dovrà essere sommato all'indirizzo di base della libreria.

Il risultato così ottenuto è l'indirizzo della jump table che contiene l'indirizzo iniziale della decima funzione della libreria in questione che a questo punto potrà essere eseguita Il tutto non è, comunque, così complesso come può sembrare da questa spiegazione; nella figura 5.3 è riportato schematicamente l'esempio appena citato.

Task

Una delle caratteristiche più esclusive del sistema operativo di Amiga è la capacità di far girare e gestire più programmi indipendenti tra di loro. Questi task sono i normali device driver di sistema, le utilities che girano in background e le normali applicazioni.

Una gestione così completa dei task è dovuta alle routine di libreria dell'Exec, che permettono di creare e rimuovere i task, di fare la schedulazione (scheduling), di gestire i segnali di evento, gli interrupt e la mutua esclusione.

Le librerie dell'Exec di Amiga offrono un ambiente operativo in real time, basato su messaggi e in multitasking:

- *real time*
  significa che le routine in ROM sono in grado di rispondere agli eventi appena essi accadono; significa inoltre che le routine di sistema sono state ottimizzate per fornire risposte più velocemente possibile;
- *multitasking*
  significa che diversi task possono operare contemporaneamente nella memoria di Amiga, senza che ciascuno di questi sia consapevole della presenza degli altri
- *basato su messaggi*
  significa che l'intero sistema operativo è stato progettato per funzionare sulla base di messaggi passati tra i vari task.

La condivisione dei task della famiglia 68000 è gestita dell'Exec; tutti i task che girano hanno un proprio stack e pieno accesso a tutti i registri. I task di sistema e le applicazioni dell'utente (che sono solitamente task di alto livello, chiamati *processi*) girano tutti in multitasking e vengono gestiti dall'Exec nello stesso modo.

I processi sono task creati tramite la libreria del dos e sono in grado di utilizzare i file per l'Input/Output, i device, le librerie residenti su disco e i font; tutti i programmi lanciati dal Workbench e dal CLI sono processi.

La schedulazione

L'Exec gestisce il multitasking suddividendo l'accesso al 68000 tra i vari task. Ciascuno di questi possiede una propria priorità e la loro schedulazione avviene in base al valore della priorità.
Il task con la priorità più alta è quello nello stato *ready* (si veda il paragrafo "Stato dei task"). Sarà questo task ad essere eseguito dal processore fino quando non si verifica uno dei seguenti eventi:

- un task con priorità superiore non diventa attivo;
- il task attuale deve aspettare un evento esterno;
- il task attuale supera il tempo massimo previsto per la sua esecuzione ed esiste un altro task pronto per essere eseguito.

Il task che sta girando può perdere l'accesso al processore in qualsiasi momento quando è necessario eseguire un altro task più urgente; quando quest'ultimo task sarà finito, il task che è stato sospeso viene ripreso dal punto in cui era stato interrotto.

Inoltre, se vi sono due task con la stessa priorità, interviene il fattore tempo: in questo caso un task viene eseguito per un certo periodo di tempo dopodiché viene sospeso per l'esecuzione del task con la stessa priorità.

Attesa

Dato che la caratteristica peculiare della schedulazione dei task è quella di essere basati sulle priorità, i task devono evitare di utilizzare la tecnica di polling per l'attesa.

In questa tecnica un pezzo di codice cicla all'infinito fino a che non si verificano cambiamenti dello stato di alcune condizioni. L'utilizzo di questa tecnica spreca molta potenzialità del processore. Inoltre, nella maggior parte dei casi, questa tecnica impedisce ai task con priorità più bassa di essere eseguiti e fa sprecare molto "tempo macchina" per task con la stessa priorità. Al contrario, utilizzando questa tecnica con task a priorità alta si rischia di non far eseguire routine di sistema urgenti;

inoltre, questa tecnica può anche causare dei blocchi del sistema.

In alternativa al polling, l'Exec fornisce una serie di routine che permettono di attendere eventi esterni senza che venga sprecato del tempo macchina.

Lo stato dei task

Per ogni singolo task, l'Exec possiede tutte le informazioni necessarie per identificare lo stato corrente, che può essere uno dei seguenti:

- running — questo stato indica che il task in quel momento viene eseguito dal processore. È anche possibile che questo task sia stato temporaneamente sospeso a causa di un interrupt di sistema;
- ready — un task in questo stato non è in esecuzione, ma è pronto per essere eseguito. Esso si trova nella lista dei task da eseguire e verrà eseguito in base alla sua priorità, relativa al task in esecuzione e agli altri task nello stesso stato;
- waiting — indica che questo task è in attesa di un evento esterno. I task in questo stato non sono schedulati, ossia non sono nella lista dei task pronti da eseguire; questi task verranno messi in questa lista solo dopo che si sia verificato l'evento atteso.

Inoltre, esistono alcuni stati transitori:

- added — indica che il task in questione è appena stato aggiunto (added) e non è ancora stato schedulato per l'esecuzione;
- removed — un task in questo stato è appena stato rimosso. Questo task è effettivamente terminato e stanno per essere rimosse tutte le strutture relative;
- exception — indica che il task è schedulato per per l'esecuzione di una exception. Una exception è un task che interrompe

tutti i task in esecuzione (qualsiasi cosa essi stiano facendo); per questo motivo vengono utilizzate solo da programmatori esperti.

La coda dei task

Prima di introdurre questo veloce argomento è necessario spiegare brevemente cos'è una coda (queue).

Le code sono semplicemente liste ordinate secondo un certo criterio.

I task che non sono nello stato di esecuzione (running) sono posti in due code di sistema. La prima (ready queue) contiene tutti quei task che sono pronti per essere eseguiti (ready): essa è costantemente mantenuta ordinata a seconda della priorità dei task: il primo elemento è il task con priorità più alta, mentre l'ultimo quello che priorità più bassa.
La seconda coda (waiting queue) contiene quei task che sono in attesa (waiting) di un evento esterno. Questa coda, al contrario di quella dei task in stato ready, non è ordinata a seconda della priorità dei task. Quando un task passa in questo stato esso viene semplicemente aggiunto come ultimo elemento della coda. I task di questa coda vi rimarranno fino a quando non si verificherà l'evento esterno che ciascuno aspetta.

Le priorità

La priorità di un task indica l'importanza del task stesso, in relazione agli altri task. I task con priorità più alta verranno eseguiti prima di quelli con priorità minore.

La priorità di un task è memorizzata come valore compreso nell'intervallo +128 e -127; la priorità più alta è rappresentata da valori positivi. Solitamente i task di sistema hanno una priorità compresa tra +20 e -20, mentre le varie applicazioni hanno priorità zero.

I segnali

I task molto spesso hanno la necessità di essere in comunicazione e in coordinazione con gli altri task o con le altre attività del sistema. Questo

coordinamento tra i vari task avviene attraverso lo scambio sincronizzato di specifici indicatori, chiamati *segnali*. Questi sono il meccanismo primario dell'interscambio di informazioni tra i vari task per una corretta comunicazione e sincronizzazione.

Il meccanismo dei segnali opera a basso livello ed è stato progettato per ottenere alte prestazioni dal sistema; esso lavora a basso livello, tanto che il normale utente non ne sospetta nemmeno l'esistenza. I messaggi di sistema (si vedano i paragrafi successivi), per esempio, utilizzano i segnali per comunicare l'arrivo di un nuovo segnale.

Il sistema dei segnali è stato progettato per supportare eventi simultanei in maniera indipendente. I segnali possono essere pensati come viaggianti in parallelo. Ogni task possiede fino a 32 segnali indipendenti che vengono memorizzati come singoli bit. Questi segnali possono verificarsi contemporaneamente.

Tutti questi segnali sono considerati relativi ai task. Un task può assegnare un significato particolare a ogni segnale, in quanto il significato dei segnali non è comune a tutti i task. Un segnale ha significato solo per il task che lo ha definito e i task lo utilizzano devono sapere quale significato ha.

L'esclusione

In tutti i sistemi multitasking si verifica la necessità di condividere i dati indipendentemente dal numero di task in esecuzione. Se i dati sono statici (ossia non cambiano mai) allora non vi è nessun problema. Se invece i dati sono dinamici (ossia sono variabili e quindi possono cambiare inaspettatamente) allora non bisogna far cambiare i dati mentre un task vi sta accedendo.
Queste strutture dati sono condivise dal sistema e dagli altri task; per potervi accedere è necessario informare il sistema e gli altri task che si stanno utilizzando queste strutture, in modo che essi non possano accedervi, modificando strutture dati che poi potrebbero mandare in crash il sistema.

Questa operazione può essere fatta in due modi: proibendo l'accesso al sistema (forbidding) o disabilitandolo (disabling) oppure utilizzando i semafori (si vedano i prossimi paragrafi).

Forbidding

Questa tecnica è utilizzata quando un task deve accedere a strutture condivide alle quali possono accedere contemporaneamente altri task. In pratica questa tecnica impedisce agli altri task di accedere contemporaneamente alle strutture dati di sistema.

Questa operazione ha lo stesso effetto che si otterrebbe disabilitando il multitasking fino a che il task rimane nello stato di esecuzione (running) In questo stato si possono verificare interrupt, ma non può essere creato o eseguito nessun task, indipendentemente dalla sua priorità.

Disabling

Questa tecnica è molto simile a quella vista precedentemente, con la differenza che vengono inibiti anche gli interrupt durante questa fase. Si ricorre a questa operazione quando si deve accedere a strutture dati condivise anche con le routine di interrupt. In pratica viene eliminata la possibilità che una routine di interrupt acceda a delle strutture dati condivise impedendo il verificarsi della richiesta di interrupt.

I semafori

I semafori possono essere utilizzati per ottenere l'esclusione dell'accesso a strutture dati condivise. Con questo metodo i task rispettano delle convenzioni comuni prima di accedere a strutture dati condivise. In pratica i task si accordano in modo di non accedere contemporaneamente. I task che non hanno necessità di accedere a questi dati continuano a girare tranquillamente, mentre viene inibito l'accesso solo a quei task che potrebbero accedervi (nei due metodi visti precedentemente, tutti i task venivano bloccati). Per questo semplice ma importante motivo, i semafori sono preferiti alle tecniche di forbidding e disabling.
I semafori sono una caratteristica importante dell'Exec; essi forniscono un metodo generale ai task per accedere alla memoria e alle strutture dati del sistema che possono essere condivise.

Per esempio, per aggiungere un nodo a una lista, un task deve semplicemente... aggiungere un nodo. Se invece la lista in questione è condivisa con altri task, questa operazione potrebbe essere dannosa. Un altro task potrebbe accedere alla lista mentre si stanno aggiornando i dati e potrebbe ricevere un puntatore errato. Il problema si verifica se i due task tentano di aggiungere contemporaneamente un nuovo elemento alla lista.

Il semaforo è simile a una chiave che serve per accedere a un dato chiuso dietro una porta. Quando si possiede questa chiave (il semaforo) è possibile accedere ai dati senza preoccuparsi che gli altri task accedano agli stessi dati. Tutti gli altri task che tentano di avere il semaforo vengono messi in attesa fino a che il semaforo non sarà nuovamente disponibile. Quando si ha finito di lavorare con questi dati si DEVE restituire il semaforo.

Per lavorare correttamente con i semafori, vi sono due restrizioni alle quale è obbligatorio attenersi:

1) tutti i task che accedono a strutture dati condivise con altri task e protette da un semaforo devono sempre chiedere il semaforo prima di accedere ai dati. Se qualche task accede direttamente ai dati senza aver ottenuto precedentemente il semaforo, i dati potrebbero essere danneggiati.
2) si può verificare una situazione di stallo se un task proprietario di un semaforo relativo a una certa struttura dati chiama un altro task che deve accedere, attraverso il semaforo, alla stessa struttura dati.

Messaggi e porte

Affinché sia possibile la comunicazione tra i diversi task, l'Exec dispone di un potente meccanismo di interscambio di dati: i messaggi e le porte. Questo meccanismo è utilizzato per passare dati di grandezza arbitraria da task a task, da interrupt a task e da task a interrupt software. Inoltre, i messaggi sono spesso utilizzati per coordinare le operazioni di cooperazione tra i vari task.

La struttura dati di un messaggio è suddivisa in due parti: i collegamenti con il sistema e il corpo del messaggio. I collegamenti al sistema vengono utilizzati dall'Exec per collegare il messaggio dato alla sua destinazione; il corpo del messaggio contiene i dati veri e propri: la sua grandezza non può superare i 64 KByte.

I messaggi sono sempre inviati a una destinazione predeterminata: la porta. Alla porta, il messaggio appena arrivato viene posto in una coda che utilizza la tecnica FIFO (First-In First-Out, ossia il primo messaggio che arriva è il primo a essere analizzato). Non vi sono restrizioni al numero di porte o al numero di messaggi che possono essere accodati a una porta (a parte la disponibilità di memoria).

I messaggi sono sempre accodati per riferimento. Per ragioni di prestazioni non viene effettuata la copia del messaggio. In pratica, un messaggio tra due task è una temporanea licenza, per il task ricevente, di utilizzare una parte della memoria del task che invia il messaggio, che è poi la parte di memoria che occupa il messaggio. Questo significa che se il task A spedisce un messaggio al task B, il messaggio fa ancora parte del contesto del task A. Il task A, comunque, non può accedere al messaggio fino a che il task B non risponde, ossia non rispedisce indietro il messaggio.

Input/Output

Uno dei principali scopi dell'Exec è quello di fornire uno standard per la gestione di tutti i device di Input/Output. Questo consiste nella definizione di un'interfaccia di device standard, di un formato per le richieste di I/O e la definizione di regole per l'interazione tra device e task. Nella progettazione del sistema di I/O di Amiga, è stata posta molta cura per evitare differenti personalizzazioni di vari device.

Un device è in realtà un'astrazione che rappresenta un set di interazioni ben definite con diversi dispositivi fisici. Questa "astrazione" è costituita da una struttura dati standard dell'Exec e da un modulo di codice di

sistema indipendente. La struttura dati fornisce l'interfaccia esterna e mantiene aggiornato lo stato corrente del dispositivo, mentre il modulo di codice effettua le operazioni necessarie per far funzionare il device. Questo modulo di codice è solitamente chiamato device driver.

Un device unit è un esempio di device. Esso condivide la stessa struttura dati e lo stesso modulo di codice con le altre unit dello stesso device, anche se operano in maniera totalmente indipendente. Spesso le unit corrispondono a sottosistemi fisici separati dello stesso genere; un esempio è quello dei floppy disk drive di Amiga che sono unit indipendenti dello stesso device: esiste solo una struttura dati e un solo modulo di codice che gestisce queste unità.

Si conclude con questo argomento il capitolo dedicato all'Exec. Naturalmente gli argomenti da trattare non sarebbero finiti, e comunque quelli affrontati sono stati trattati il più superficialmente possibile (senza però omettere caratteristiche e funzioni essenziali).

Per un approfondimento di questi argomenti si consiglia di leggere manuali specializzati dedicati alla struttura interna di Amiga.

Dal prossimo capitolo, invece, verranno trattati argomenti "software", iniziando, ovviamente, dal sistema operativo.

Capitolo 6

AMIGADOS 1.3: IL CLI

In questo capitolo verrà trattato l'AmigaDOS, che, come ripetuto più volte, è il sistema operativo multitasking che consente di far girare contemporaneamente più programmi. Viene analizzata la release 1.3 del sistema operativo, in quanto è la più diffusa: le versioni 1.1 e 1.2 non vengono trattate in quanto obsolete (la maggior parte dei programmi richiede come minimo la versione 1.3 per poter girare), mentre la versione 2.0 verrà trattata nei capitoli successivi.

Le due interfacce

L'AmigaDOS è l'unico sistema operativo per personal computer ad utilizzare due tipi di interfaccia: quello a linea di comando (CLI, Command Line Interface) e quello a icone (Workbench) che verrà trattato nel capitolo successivo. Il CLI è gestibile solo ed esclusivamente attraverso tastiera, mentre il Workbench è, come appena accennato, il sistema a icone e finestre che è ovviamente comandabile solo tramite mouse.

Il CLI

Tramite CLI è possibile effettuare tutte le operazioni possibili per una gestione completa dei file. Prima di iniziare a parlare dei comandi del CLI vediamo brevemente come sono organizzati e divisi i file sui floppy disk.

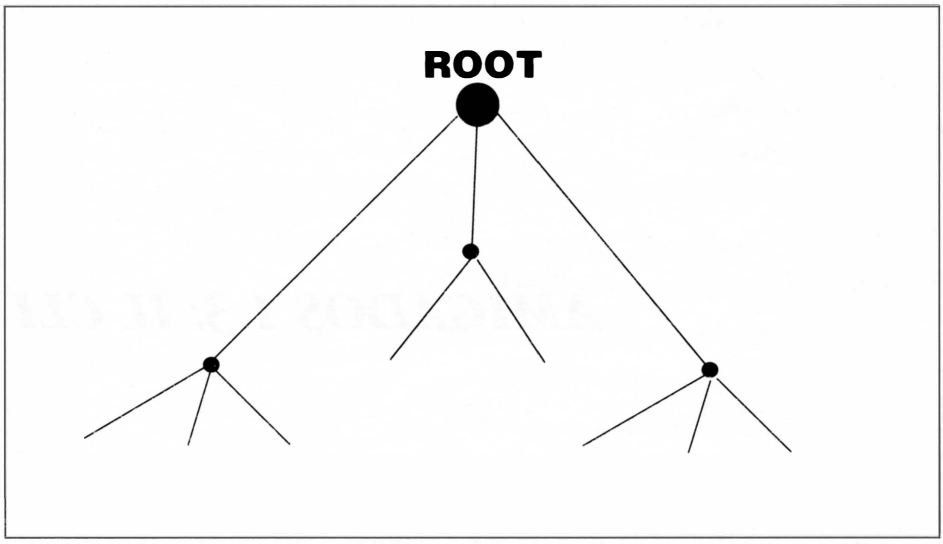

Figura 6.1
La struttura gerarchica ad albero

Innanzitutto i file sono organizzati in maniera gerarchica, ossia su tutti i dischetti esiste un'organizzazione dei file come quella mostrata in figura 6.1. Come si vede esiste una base (root, radice) da dove partono delle diramazioni: per questo motivo l'organizzazione dei file viene spesso definita ad albero (tree).

Per spiegare meglio l'organizzazione dei dischi immaginiamo che ogni disco sia un armadio che contiene i nostri documenti: esso può contenere in maniera confusa tutti i nostri documenti (figura 6.2 a) oppure dividerli molto più logicamente per argomenti in appositi cassetti (figura 6.2 b); questi cassetti sono chiamati directory: solitamente ogni directory contiene file dello stesso argomento, come possono essere i comandi CLI, le librerie o i nostri disegni.

Per accedere al CLI (o Shell) è necessario premere contemporaneamente i tasti CTRL-D durante il caricamento del dischetto (fase detta di bootstrap) oppure, da Workbench, cliccare due volte sull'icona CLI o Shell. Una volta entrati in CLI ci troviamo il prompt con il numero del CLI:

```
1>
```

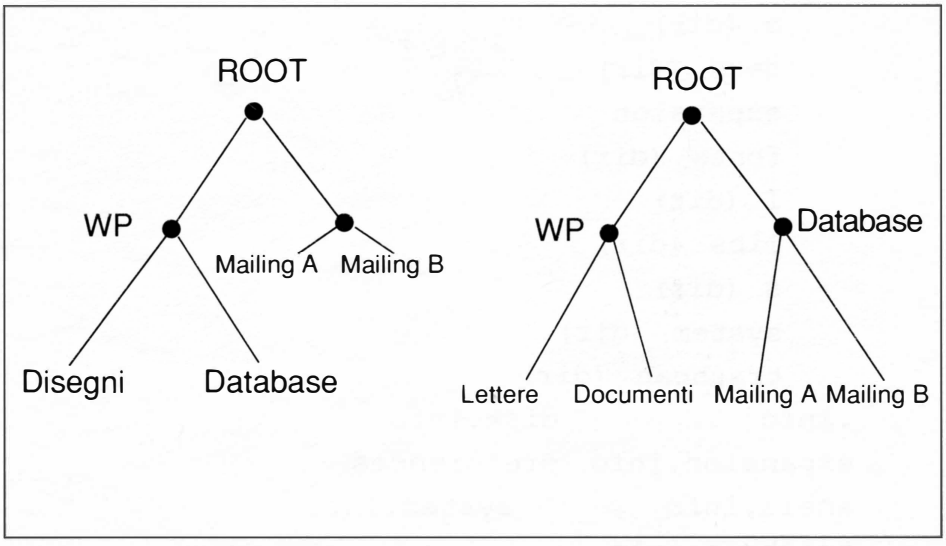

Figura 6.2
Una struttura disordinata e una divisa in directory

In questo caso il prompt ci informa che il processo attuale è il numero 1; è possibile però che il prompt si presenti sotto questa forma:

```
1 PIPPO:S>
```

Questa volta il prompt indica oltre al numero del processo anche la directory attuale nella quale ci troviamo; nel caso appena esposto ci si trova nella directory S del disco denominato PIPPO.

Vediamo ora i primi comandi che ci permettono di vedere il contenuto di una directory e di cambiare directory.

Il primo comando che tutti imparano è DIR, il cui compito è quello di mostrare il contenuto di una directory. Dal prompt bisogna digitare

```
1.PIPPO:> dir
```

appariranno all'incirca i seguenti file:

```
    c (dir)
    devs (dir)
    expansion
    fonts (dir)
    l (dir)
    libs (dir)
    s (dir)
    system (dir)
    trashcan (dir)
.info           disk.info
expansion.info  preferences
shell.info        system.info
trashcan.info
```

Queste sono le directory che solitamente compongono il dischetto di sistema del Workbench; le voci che a lato hanno "(dir)" sono directory, mentre tutti i file che hanno estensione .info sono quelli visibili nelle finestre del Workbench: per esempio disk.info contiene l'immagine del dischetto, system.info contiene l'immagine di un cassetto (per rappresentare una directory) e trashcan.info rappresenta la directory trashcan con un'immagine di un cestino (trashcan).

La voce preferences è un programma che permette di impostare i parametri del sistema, come i colori, la forma del puntatore del mouse o la stampante da utilizzare.

Nella directory "c" sono contenuti tutti i comandi dell'AmigaDOS. Per vederli ci sono due modi. Il primo è quello di utilizzare il comando dir seguito dal nome della directory che si vuole analizzare:

```
1.PIPPO:> dir c
```

Il secondo modo è, invece, quello di impostare come directory attuale la directory c. In questo modo ci si posiziona in quella directory e

digitando dir si ottiene l'elenco dei file contenuti nella directory attuale. Il comando che permette di spostarci in un altra directory è CD (Change Directory) seguito dal nome della directory in cui ci si vuole posizionare.

```
1.PIPPO:> cd c
1.PIPPO:C> dir
```

I comandi elencati non sono altro che veri e propri programmi che svolgono piccole funzioni come copiare file da una parte all'altra dello stesso disco o da un disco all'altro, cambiare nome ai file, inizializzare o formattare interi dischi e molto altro ancora. Se invece volevamo analizzare il contenuto della directory "printers" che si trova a sua volta all'interno della directory "devs" anziché digitare:

```
1.PIPPO:> cd devs
1.PIPPO:devs> cd printers
1.PIPPO:devs/printers>
```

è possibile inglobare tutto in un unico comando, unendo le directory con il "/"

```
1.PIPPO:> cd devs/printers
1.PIPPO:devs/printers>
```

A questo punto, dopo aver visto il contenuto della directory, supponiamo di voler tornare alla radice del disco (root): anche in questo caso vi sono due modi. Il primo è quello di utilizzare la barra retroversa:

```
1.PIPPO:devs/printers> cd /
1.PIPPO:devs> cd /
1.PIPPO:>
```

Il secondo è quello di utilizzare il carattere due punti ":" che indica la root del disco da qualsiasi punto del disco ci si trovi

```
1.PIPPO:devs/printers> cd :
1.PIPPO:>
```

Dal CLI è possibile eseguire programmi applicativi come word processor, spreadsheet o programmi di grafica. Per fare ciò è necessario che essi si trovino nella stessa directory dalla quale li si lancia, o, per meglio dire, è necessario portarsi nella directory in cui si trova il programma. Per esempio se ci si trova nella directory devs/printers e si vuole eseguire il programma preferences...

```
1.PIPPO:devs/printers> preferences
Bad command or file name
1.PIPPO:devs/printers>
```

... si otterrebbe un messaggio di errore che ci informa che il programma desiderato non esiste nella directory attuale. Per ovviare al problema è semplicemente necessario portarsi nella directory precedente tramite "cd :" oppure digitare :preferences per indicare che il programma si trova nella root.

A questo punto ci si pone un quesito: abbiamo detto che i comandi CLI non sono altro che programmi; abbiamo appena detto che i programmi per essere eseguiti devono stare nella stessa directory dalla quale li si lancia: allora, perché i comandi CLI vengono eseguiti anche se non ci si trova nella stessa directory?

La risposta a questo quesito non è difficile; l'AmigaDOS, quando viene impartito un programma, va a cercarlo nella directory attuale (come già sappiamo) e, se non viene trovato, va a cercarlo in una serie di directory specificate dal comando PATH (percorso). Per default, il percorso di ricerca viene inizializzato alla directory c del disco di avvio. Questo è il motivo per cui i comandi del CLI vengono eseguiti. Digitando PATH al prompt dovremmo ottenere all'incirca...

```
1.PIPPO:devs/printers> path
```

```
Current directory
C:
1.PIPPO:devs/printers>
```

Come si può notare è apparso un C: che sta a indicare un *device logico*; facciamo un passo indietro per spiegare cos'è un device logico.

Dal momento che tutto il lavoro di Amiga richiede sempre alcuni file come librerie, device driver o comandi CLI, i programmatori di Amiga hanno deciso di individuare questi file tramite un'etichetta, in modo che tutti i programmi, quando hanno bisogno di un determinato file utilizzano questa etichetta per individuarlo. La directory "c" che contiene tutti i comandi CLI è identificata come C:, così come la directory LIBS, che contiene le librerie, è identificata come LIBS:. Questi device logici vengono assegnati automaticamente all'avvio del sistema alle directory del disco di avvio.

Torniamo al nostro discorso. Se volessimo aggiungere una nuova directory basta digitare il comando PATH seguito dal nome della directory nella quale si vuole far eseguire la ricerca:

```
1.PIPPO:devs/printers> path c:/system
1.PIPPO:devs/printers> path
Current directory
C:
PIPPO:system
```

Suponiamo ora di voler controllare il contenuto di un dischetto; togliamo il dischetto attuale dal drive df0: con il quale abbiamo eseguito il boot e inseriemo il dischetto da controllare. Digitiamo dir e si aprirà una piccola finestrella in alto a sinistra dello schermo con il seguente messaggio

```
Please insert disk
PIPPO
in any drive
```

Cosa è successo? Come abbiamo appena detto l'AmigaDOS ricerca il comando impartitogli nella directory attuale che nel nostro caso si trova sul disco PIPPO. La finestrella che è apparsa è un classico messaggio di sistema che richiede all'utente di compiere qualche operazione per portare a buon fine l'operazione richiesta; questi messaggi sono detti *System request*, come indica il titolo della finestra. In pratica, per poter visualizzare il contenuto di un altro disco, è necessario avere il comando dir sotto mano; si può procedere in tre modi:

1) copiare nella *RAM Disk* il comando dir (e tutti quelli che servono), e assegnare alla RAM Disk il device logico C: tramite il comando *assign*; la Ram Disk è a tutti gli effetti un disco (il cui nome è RAM:) che si trova nella memoria interna di Amiga;
2) utilizzare il comando resident;
3) utilizzare un disco di lavoro.

Il primo metodo era quello utilizzato fino alla versione 1.2, in quanto non esisteva il comando resident; per effettuare la copia dei comandi in RAM: è necessario utilizzare il comando copy:

```
1.PIPPO:devs/printers> makedir RAM:C
1.PIPPO:devs/printers> copy C:  RAM:C
1.PIPPO:devs/printers> assign C: RAM:C
```

In questo modo si è creata una directory nel disco RAM: di nome C, si è copiato il contenuto del device logico C: nella nuova directory appena creata e gli si è assegnato il device logico che prima apparteneva alla directory c del disco di avvio. Naturalmente, chi ha solo 512 KByte di RAM, dovrà copiare solo i comandi più utilizzati.

Il secondo metodo, come abbiamo detto, utilizza il comando resident che non fa altro che copiare in memoria il comando che viene specificato di seguito. Quando si digita un comando, esso viene innanzitutto cercato nella lista dei comandi residenti in memoria e successivamente viene utilizzata la lista del percorso di ricerca.

Il terzo metodo consiste nel creare un disco di lavoro, ossia un disco che sia in grado di avviare il sistema (fase di boot-up) con solo i comandi necessari e alcune utility, ed è attuabile solo se si posseggono due drive: nel drive df0: si inserirà il disco appena creato, e nel secondo drive il disco da analizzare.

Passiamo ora alle altre directory del disco Workbench. Nella directory "devs" si trovano i file *.device* (serial, parallel, ecc) che contengono le istruzioni per interfacciarsi con dispositivi esterni.

Nella directory fonts si trovano i font (caratteri) utilizzati da Amiga; all'interno di questa directory si trovano un certo numero di file con altrettante directory che hanno i nomi dei font (per esempio, topaz, shappire, ruby, ecc) e che contengono la descrizione in punti dei caratteri: all'interno della directory topaz troveremo tre file "8", "9" e "11", ciascuno dei quali contiene la rispettiva descrizione bitmap dei caratteri.

I file di cui si parlava precedentemente hanno come nome quello dei font, con il suffisso ".font" e contiene informazioni di carattere generale, come, per esempio, se il font è proporzionale e in che directory si trovano le descrizioni bitmap delle grandezze di font disponibili.

Se, per esempio, nella directory font si trova solo il carattere topaz, digitando "dir fonts" otterremo:

```
1.PIPPO:> dir fonts
          topaz (dir)
     topaz.font
1.PIPPO:>
```

Un'altra directory che riveste particolare importanza è quella che contiene le librerie (si veda il capitolo relativo all'Exec), la directory "libs".

All'interno di questa directory non si trovano però tutte le librerie di Amiga, in quanto le più importanti, nonché utilizzate sono residenti in

ROM (ossia nel Kickstart); in questa directory, infatti si trovano le librerie meno utilizzate, come le librerie matematiche.

La directory "s" contiene tutti i file script (s sta appunto per script). Un file script (o un file batch, che è la stessa identica cosa) è un file di testo contenente una serie di comandi CLI che permettono di eseguire operazioni di routine. La stessa startup-sequence (il file che viene eseguito automaticamente nella fase di boot del sistema) è un file script e contiene tutti i comandi che servono per inizializzare il sistema (l'impostazione del percorso di ricerca dei comandi, l'impostazione della data e dell'ora, per chi non possiede la batteria tampone, il caricamento del Workbench, ecc).

L'AmigaDOS possiede un discreto set di comandi esclusivamente dedicato alla programmazione batch, come le corrispondenti istruzioni BASIC "if...then...else" o come l'istruzione BASIC "goto".

Capitolo 7

IL WORKBENCH

Il Workbench (in italiano è approssimativamente traducibile con "banco di lavoro") è sicuramente lo strumento che mette più direttamente a contatto Amiga con l'utente. Con l'aiuto del Workbench e le facilitazioni fornite da Intuition al Workbench, Amiga diventa quello che oggi è conosciuto come ambiente "WIMP", un ambiente dove gli utenti lavorano con finestre, icone, un mouse e un puntatore. Il Workbench è quindi un programma che permette di amministrare Amiga, e i programmi che girano su di esso.

Caratteristiche del Workbench

Osservando il Workbench così come appare sullo schermo si possono notare alcune caratteristiche:

Il cursore

Nello schermo, appena caricato il dischetto, esiste una piccola freccia: il cursore del mouse o puntatore. Quasi tutto ciò che si esegue da Workbench dipende dalla posizione di questo puntatore.

La barra del menu

Per facilitare l'utente sono stati inclusi anche i menu, liste delle potenzialità d'azione dei programmi. Si può accedere ai menu, listati nella sbarra in cima alo schermo, tenendo premuto il pulsante destro del mouse,

benché all'esterno di un particolare programma i contenuti della barra possono cambiare se ci si muove tra differenti parti del programma.

Le finestre

Una delle limitazioni dei computer non WIMP è l'impossibilità di mantenere sullo schermo due display diversi, l'ultimo cancella il precedente. Con Amiga esistono le finestre, dunque è possibile vedere contemporaneamente, ad esempio, il contenuto di più dischetti.

Le icone

La parola icona, che significa immagine, si riferisce al piccolo disegno che appare sul Workbench rappresentante le scelte, item, che il sistema permette di controllare.

Tecniche per utilizzare il Workbench

Avendo delineato, per grandi linee, gli elementi che compongono il Workbench, proviamo a fare un po' di esperienza, naturalmente consigliamo di caricare una copia del dischetto Workbench per prevenire eventuali perdite irreparabili, con lo stesso.

Selezioni

Per selezionare un elemento è sufficiente porre il cursore sopra l'elemento prescelto e premere il pulsante sinistro del mouse: a questo punto lo status dell'elemento cambierà, ad esempio l'icona cambierà aspetto.

Il doppio click

Per avviare un programma o vedere il contenuto di un disco, basta porre, naturalmente, il cursore sopra l'elemento e premere due volte successivamente, molto velocemente, il pulsante del mouse.

Il trascinamento

Per muovere un elemento, cioè "trascinare", bisogna "selezionare" un elemento ma non rilasciare il pulsante sinistro del mouse; spostando il mouse si sposterà anche l'elemento prescelto.

Selezioni multiple

Se un processo richiede selezioni multiple bisogna eseguire lo stesso procedimento della Selezione e tenere premuto assieme il tasto shift. Per deselezionare tutti gli elementi è sufficiente cliccare su una parte vuota del Workbench. Invece, per deselezionare un solo elemento di un insieme, bisogna selezionarlo una seconda volta. Inoltre, è possibile trascinare tutti gli elementi selezionati.

Gestione delle finestre

Apertura: le finestre sono aperte premendo due volte sull'icona di un disco o di un drawer (cassetto). Un metodo alternativo è selezionare un elemento e utilizzare la voce OPEN sul menu Workbench; vale anche la multi-selezione.

Movimento: per muovere una finestra, dopo averla aperta, bisogna portare il puntatore sulla sbarra superiore della finestra e premere il pulsante sinistro del mouse; si può così muovere su tutto lo schermo la finestra, anche se nessuna sua parte può uscire dai bordi dello schermo.

Selezione: prima di poter eseguire un'azione sugli elementi di una finestra bisogna selezionarla. Una finestra selezionata è riconoscibile osservando la sbarra trasversale sul bordo alto, che nella finestra selezionata è compatta piuttosto che ombrata. Per selezionare su una finestra basta posizionare il cursore sulla finestra e cliccare il tasto sinistro del mouse.

Dimensionamento: un'operazione utile che si può effettuare su una finestra, è alterare le sue dimensioni. Il dimensionamento si ottiene portando il cursore sul simbolo in basso a destra della finestra prescelta e premere il pulsante sinistro del mouse e, naturalmente, spostare il mouse. La dimensione massima è uguale all'intera dimensione dello schermo.

Scorrimento: quando una finestra non rivela l'intero contenuto di un disco perché tale contenuto è troppo grande per essere posto tutto in una finestra, è possibile muovere la finestra sul contenuto o, effettivamente, muovere il contenuto della finestra affinché si possano vedere gli elementi che prima erano nascosti. La finestra può essere mossa solo se

vi è qualcosa che non è visibile. Per determinare se ci si trova in questo caso, si devono guardare le sbarre di scorrimento la barra destra e inferiore che limitano la finestra). Se queste sono completamente bianche significa che tutto è stato visualizzato. Esistono due tipi di movimento di una finestra:

1) la freccia delle sbarre di scorrimento.
2) la sbarra di scorrimento.

Chiusura di una finestra: quando si ha finito di usare una finestra e si desidera toglierla dallo schermo, si seleziona semplicemente il simbolo di chiusura posto nell'angolo in alto a sinistra.

Come portare una finestra su uno schermo

Lavorando con le finestre non è necessario ridurre la dimensione di ciascuna per fare lo spazio alle altre. In molti casi è conveniente tenere la finestra della stessa dimensione e utilizzare i gadget di "profondità" posti nell'angolo destro in alto alto della finestra. Questi due gadget permettono di sovrapporre o nascondere una finestra.

L'indicatore di completo/vuoto

Un tipo di finestra possiede un gadget speciale, a sinistra della finestra, che indica l'ammontare dello spazio libero su disco.

I menu del Workbench

Ricordando le tecniche dei menu, diamo un velocissimo sguardo ai menu stessi.

Il menu "Workbench"

Open

Open è l'alternativa alla doppia pressione del pulsante su un'icona. Se si seleziona Open l'icona selezionata sarà attivata come se si fosse premuto due volte il pulsante su di essa.

Close

Consente di chiudere la finestra attualmente selezionata; ha lo stesso effetto del click sul gadget di chiusura della finestra.

Duplicate

Va utilizzata per creare una copia di un disk file, di un programma o di un drawer. Per duplicare un singolo elemento (selezioni multiple non sono permesse), bisogna selezionare l'elemento e poi attivare l'opzione Duplicate. Questa opzione è necessaria solo quando si desidera duplicare un elemento sullo stesso disco.

Rename

Permette di specificare un nome diverso alla icona selezionata, sia essa un disco, un file o altro. Per assegnare un nuovo nome, bisogna selezionare un elemento e poi richiamare l'opzione. Un requester apparirà affinché si specifichi il nuovo nome da assegnare all'elemento.

Info

Questa opzione permette all'utente di visualizzare e alterare un certo numero di significative caratteristiche di un file. Per visuliazzare tali informazioni, è sufficiente selezionare l'icona e andare sull'opzione Info. Le principali caratteristiche sono:

a) il nome del file, tipo e ampiezza.
b) le dimensioni dello stack.
c) lo status di un file.
d) il commento.
e) il default tool.
f) il tool types.

Discard

Questa opzione cancella irrimediabilmente un drawer o un file selezionato dal disco. Discard può essere utilizzato con selezioni multiple.

Il menu "Disk"

Empty Trash

Questa opzione cancella il contenuto del Trashcan (il cestino).

Initialize

Quando un nuovo disco è inserito per la prima volta nel sistema non può essere letto o scritto. Ciò non solo perché non c'è nulla nel disco, ma soprattutto perché manca delle strutture di cui Amiga necessita.
Dunque va inizializzato, o formattato: ciò avviene mettendo il disco nel drive, selezionando la sua icona e poi usando l'opzione Initialize.

Il menu "Special"

Clean Up

Con questa opzione è possibile permettere al sistema di fare un veloce riordinamento del materiale cosicché sembri più ordinato sullo schermo, selezionando l'icona di un disco o di un drawer e poi richiamando questa opzione. L'ordinamento non è permanente finché l'opzione Snapshot è forzata sulle icone che sono state ordinate.

Last error

Mostra sullo schermo l'ultimo messaggio d'errore apparso.

Redraw

L'opzione Redraw richiede a Intuition di ricreare il display riparando così ogni danno causato da qualche programma.

Snapshot

Per registrare un cambiamento permanente sul disco si deve usare questa opzione, ad esempio dopo aver riordinato una finestra oppure dopo aver ridimensionato uno schermo o, ancora, dopo averlo spostato.

Version

L'opzione Version mostra all'utente la versione di Workbench utilizzato.

Capitolo 8

IL WORKBENCH 2.0

Pur rimanendo ancorato agli stessi principi generali di funzionamento che lo hanno da sempre caratterizzato (illustrati nel capitolo 7), il Workbench ha subito un'evoluzione parallela a quella dei sistemi operativi di Amiga, modificando e migliorando alcune sue caratteristiche. In particolare, si è avuta una consistente innovazione a cominciare dalla versione 2.0, di cui ci occuperemo in queste pagine, fornito "di serie" sugli Amiga 600, 500 Plus e 3000, ma installabile come upgrade anche nei precedenti modelli 500 e 2000.

Icone vecchie e nuove

La prima e più appariscente differenza con il vecchio 1.3, è rappresentata dalla nuova colorazione adottata. Un particolare di importanza non determinante, essendo l'impostazione cromatica modificabile con il set di programmi Preferences, ma che va citato per alcuni piccoli problemi che comporta. Il nuovo look, basato su un effetto tridimensionale dei contorni di finestre e icone, ha infatti invertito il rapporto prima esistente tra il bianco e il nero. Il che, come immediata conseguenza, porta, per esempio, a una pessima visualizzazione delle icone disegnate per essere visualizzate in ambiente 1.3.

Chi avesse cominciato la propria attività di utente Amiga prima dell'avvento del 2.0, e volesse ancora utilizzare le raccolte di icone in suo possesso senza doverle ritoccare manualmente, può comunque ricorrere a due soluzioni: adoperare alcuni piccoli programmi, circolanti negli

ambienti del pubblico dominio, che automatizzano la ridefinizione dei colori delle icone adattandole al nuovo stile, oppure modificare i colori dello schermo in modo che si adattino provvisoriamente alle vecchie colorazioni. Qualche nome non guasta: Iconverter e RemapInfo per la prima soluzione, ColorSwitch per la seconda.

Colori e Preferences

Estetica a parte, il Workbench 2.0 permette di settare per il suo schermo fino a 16 colori, adoperando uno dei programmi contenuti all'interno del cassetto Prefs: ScreenMode. Chiaro che, se si imposta questa modalità, saranno utilizzabili icone disegnate con un pari numero di colori, cosa prima possibile solo tramite utility esterne, anche se riducendo la velocità di alcune operazioni su finestre e schermi. E già che si è in argomento, proprio le Preferences costituiscono un altro elemento di novità: quello che prima era un unico programma, ora è suddiviso in più file dotati di icona.

Soffermiamoci ancora sull'aspetto estetico: si provi, a titolo di esempio, a lanciare (premendo due volte il pulsante sinistro del mouse in rapida sequenza sulla sua icona) il programma WbPattern, dopo avere aperto la finestra Prefs. Si porti ora il cursore (la freccetta) sul riquadro Presets e, al suo interno, si selezioni uno dei motivi ornamentali presenti (portandovi sopra il puntatore e premendo il pulsante sinistro del mouse). Se, a questo punto, si seleziona l'opzione Test, si vedrà lo sfondo del Workbench assumere in tutta la sua superficie il disegno prima scelto. Lo stesso procedimento può essere adottato per l'interno delle finestre, selezionando però il piccolo riquadro "Windows" in alto a sinistra. Volendo, si potrà anche disegnare un motivo di propria invenzione, sfruttando il mouse come se fosse un pennello direttamente sull'ingrandimento presente al centro della finestra.

Se lo si volesse adottare ogni volta che il sistema viene attivato, basterà agire sul riquadro Save cliccandovi dentro con il mouse.

Per un controllo generico sui colori, andrà invece usata l'icona Palette, analoga a quella già presente nei vecchi sistemi 1.3.

Si avrà modo, in seguito, di approfondire l'uso delle Preferences più sofisticate. Per ora, concludendo questa rassegna riservata all'estetica

generale, non resta che citare l'icona Fonts, anch'essa inclusa tra le Preferences. Si provi a lanciare il relativo programma e a cliccare sui colori delle due barre di scelta, evitando di associare lo stesso colore a entrambe le opzioni, e controllando che in alto sia "premuto" il pulsantino "Workbench Icon Text" e in basso a sinistra la voce "Text & Field".
Se, a questo punto, si esce dal programma adoperando il riquadro Use, si constaterà come i nomi associati alle icone avranno assunto la colorazione prima impostata.
Con l'utility Font, in realtà, si può fare molto di più, ma se ne riparlerà in dettaglio affrontando il DOS 2.0.

Le finestre

Si è già detto come l'uso generale del Workbench sia rimasto pressoché immutato, ma fa eccezione qualche dettaglio riguardante le finestre video. Intanto, quelle che vengono aperte se si seleziona un'icona-disco non sono più dotate della barra verticale che indica lo stato di riempimento del disco: l'informazione, ora, è molto più chiaramente specificata sulla barra superiore delle finestre, con un'indicazione scritta sull'esatta quantità e percentuale di Kilobyte liberi e occupati.
Inoltre, i due piccoli gadget nell'angolo in alto a destra hanno assunto una diversa funzione: uno solo (quello di destra, detto "di profondità") si occupa di portare la finestra in primo piano o dietro le altre, mentre il secondo consente di rimpicciolirla o ingrandirla con una sola manovra, senza dover utilizzare il tradizionale angolo inferiore destro della stessa.

I menu

Anche i menu della barra di schermo, quelli accessibili premendo il pulsante destro del mouse, con la versione 2.0 del Workbench sono stati parecchio modificati, tra l'altro avvicinando di molto le prestazioni ottenibili con le due diverse interfacce di Amiga: quella basata su mouse e icone, e quella (più ostica) rappresentata dalla riga di comando di una finestra Shell. Esaminiamone le principali voci, ignorando quelle rimaste immodificate (per le quali si rimanda al capitolo 7), o il cui utilizzo non coinvolge direttamente gli utizzatori del Workbench.

Il menu Workbench

Backdrop

Questa opzione funziona come un interruttore, ovvero attiva o disattiva il suo stato a seconda dell'attuale condizione. Quando è attiva, accanto alla voce Backdrop viene inserito un contrassegno. In pratica, con Backdrop il Workbench potrà essere adoperato come se si trattasse di una qualunque altra finestra, oppure in modo schermo, così come avveniva in passato.
Manipolandolo come una finestra, se ne potranno sfruttare tutti i cosiddetti gadget: regolarne le dimensioni, portarlo in primo piano o in sottofondo, e, soprattutto, farne scorrere il contenuto come più aggrada, tramite le solite freccette presenti nell'angolo inferiore destro. Può risultare comodo anche per visualizzare, senza altri elementi grafici di confusione (le icone dei dischi, della RAM disk, eccetera), la finestra di un'applicazione al momento operativa: basterà in questo caso rimpicciolire al minimo la finestra del Workbench, ed eventualmente metterla in secondo piano con il gadget di profondità.

Execute Command

E' questa una delle più grosse innovazioni, che consente di lanciare qualunque programma anche se non è predisposto all'attivazione da Workbench. Dopo avere selezionato questa voce, si disporrà di una riga di comando esattamente come se ci si trovasse in ambiente Shell. All'interno di questa, si potrà digitare qualunque istruzione del DOS: se necessario, provvederà automaticamente il sistema ad aprire una finestra di output per il programma lanciato. Per toccarne con mano l'azione, si provi a digitare "List" (senza le virgolette) nel riquadro aperto dalla funzione Execute Command, facendo seguire la pressione del tasto Invio (o Return che dir si voglia). Dopo l'elenco mostrato a video, si dovrà provvedere a chiudere la finestra creata dal sistema cliccandone come di consueto il piccolo riquadro nell'angolo superiore sinistro.
Ovvio che, per apprezzare appieno l'utilità di Execute Command, occorre una certa dimestichezza con le regole del DOS.

Update All

Rieffettua una scansione del contenuto di tutte le finestre, aggiornandolo nel caso siano state effettuate modifiche dall'esterno del Workbench, utilizzando Shell e i suoi comandi.

Last Message

Corrisponde alla vecchia opzione Last Error, che nell'1.3 faceva parte del menu Special (non più presente nel 2.0).

About...

Mostra, oltre a un messaggio di Copyright, la versione di Kickstart e Workbench presente sul sistema. In pratica, entrambe dovranno corrispondere a 37.xxx, con il numero di revisione variabile a seconda del modello e della data di acquisto.

Il menu Window

New Drawer

Con questa comoda opzione è possibile creare un nuovo cassetto, automaticamente fornito di icona, il cui nome andrà immesso in un apposito requester (si chiama così la finestra proposta da Amiga). Il cassetto verrà creato all'interno della finestra al momento attiva. Si ricordi che, con il termine attiva, si intende quella finestra il cui bordo è evidenziato dal colore blu (se non si è modificata la palette), ovvero si è cliccato una volta al suo interno.

Open Parent

Se si è aperto un "cassetto" (meglio definito con nome "directory") a sua volta contenuto in un altro, e si fosse richiuso il primo, con questa scelta lo si potrà riaprire senza doverlo di nuovo andare a cercare tra quanto contenuto in un disco o in altre directory. Il contenitore-finestra immediatamente superiore viene definito con il termine Parent, da cui il nome del menu.

Update

Funziona come Update All del menu Workbench, ma riferito alla sola finestra al momento attiva.

Select Contents

Seleziona tutte le icone presenti nella finestra attiva, esattamente come se si fosse cliccato su di esse mantenendo abbassato il tasto Shift, prima unico modo per ottenere un risultato simile. Se, per esempio, si volessero copiare tutti i programmi di una directory in un disco o RAM Disk, sarebbe sufficiente adoperare questa opzione da menu e, cliccato una sola volta su una delle icone selezionate mantenendo contemporaneamente abbassato il tasto Shift, trasportarle nella finestra (o icona) di destinazione.

Snapshot

E' suddivisa in due submenu: Window e All. La prima opzione rende permanente la dimensione e la posizione della finestra attualmente attiva, in modo che si ripresenti esattamente allo stesso modo quando la si riaprirà in futuro. In pratica, Snapshot interviene solo sul "bordo" della finestra, senza memorizzare la posizione delle eventuali icone presenti al suo interno. Con l'item All, invece, si estende la sua azione sia alla finestra in senso stretto che al contenuto (le icone al suo interno).
Una voce omonima, seppure priva dei sottomenu, è presente nel menu Icons, e svolge lo stesso compito limitatamente alla singola icona al momento selezionata (così come avveniva nel vecchio Workbench 1.3).

Show

Anche in questo caso ci troviamo di fronte a due possibilità di scelta: Only Icons e All Files. Con una rivoluzionaria innovazione rispetto al passato, nella finestra attiva possono essere visualizzati non solo i file forniti di icona (opzione Only Icons), ma anche tutti gli altri che ne sono sprovvisti (All Files). Penserà il sistema, in questo caso, a fornire di icona fittizia il file, permettendone la visualizzazione. L'icona, che comunque non sarà permanente, verrà adattata alle caratteristiche dell'unità associata:

un cassetto se si tratta di una directory, un'icona Tool se un file eseguibile, eccetera. Perché la visualizzazione avvenga, deve comunque essere attiva la voce Icon del prossimo menu di scelta.

View By

A questa voce sono associate quattro possibilità, che determinano il modo in cui verrà visualizzato il contenuto della finestra attiva: Icon, Name, Date e Size.
Con il primo modo (Icon) tutto sarà come sempre legato alle icone, mentre con gli altri tre si avrà una lista dei file presenti nella directory esattamente come verrebbe mostrata all'interno di una finestra Shell impartendo un comando List. A differenza che in ambiente Shell, resta però possibile adoperare il mouse direttamente sui nomi per aprire directory, lanciare programmi, spostare file, eccetera.
La differenza tra le tre voci che escludono le icone, consiste unicamente nell'ordinamento che assumeranno i file all'interno della finestra: alfabetico basato sul nome con l'opzione Name, in ordine di data (dalla più recente a ritroso) con Date e in base alle dimensioni con Size.
Nella pratica, un utente fedelissimo al Workbench troverà difficilmente comodo adoperare qualcosa di diverso da Icons.
Si tenga presente che le scelte legate a View By, come pure al precedente item Show, divengono permanenti (fino a succesive modifiche) quando si adopera Snapshot.

Il menu Icons

Questo menu contiene molte voci che risultano identiche a quanto già presente nella precedente versione di Workbench: Snapshot, Open, Copy, Delete (nell'1.3: Discard), Format Disk (nell'1.3: Initialize), Empty Trash, Information (nell'1.3: Info).
Ci limiteremo dunque a citare gli unici due comandi realmente innovativi, e molto comodi per chi adopera intensamente il Workbench: Leave Out e Put Away.
Come forse noto, una qualunque icona presente all'interno di una finestra può essere spostata non solo all'interno di altre, ma anche

all'esterno, direttamente nello schermo del Workbench. In tal modo, si potrà poi richiedere la finestra che la conteneva, e adoperarla tutte le volte che se ne ha bisogno. Di solito, questo tipo di operazione risulta utile quando si pensa di utilizzare con una certa frequenza quello specifico programma o directory. Lo spostamento non è comunque reale: il file verrà sempre cercato dal sistema nella sua directory (cassetto, o disco) di origine, e la sua icona resterà sullo schermo del Workbench solo fino al reset del computer.
Se, però, dopo lo spostamento, si attiva l'icona (con un solo click) e si seleziona Leave Out, questa rimarrà posizionata permanentemente sullo schermo, anche dopo un reset. In pratica, sempre pronta per l'uso. Tra l'altro, la sua esatta posizione sullo schermo può essere scelta e "fissata" con la voce Snapshot, scelta dal menu Icons.
Put Away svolge invece il compito opposto: elimina dallo schermo un'icona precedentemente fissata con Leave Out, ripristinando la sua normale posizione all'interno del cassetto o disco originario.

Il menu Tools

Questo menu, apparentemente inutile, è predisposto per essere configurato dall'esterno. Operazione, questa, che solo i programmatori più esperti sono in grado di svolgere. Esistono anche delle utility che consentono a chiunque di inserire nel menu Tools i nomi di quei programmi che si desidera attivare immediatamente, senza la necessità di ricorrere a icone, finestre, eccetera. Utile soprattutto per chi dispone di un hard disk, ma anche con i soli floppy potrebbe risultare vantaggioso un utilizzo di questo tipo. Una per tutti, citeremo l'utility Tools Daemon, reperibile presso i canali del pubblico dominio.
Va però detto che, a fronte della comodità di questi programmi rivolti al menu Tools, per il loro sfruttamento è richiesta una buona conoscenza dei meccanismi del DOS.

Capitolo 9

AMIGADOS 2.0: L'AMBIENTE SHELL

Seppure in modo meno appariscente e netto che nel contiguo universo Workbench, anche l'interfaccia Shell ha subito numerose innovazioni nel passaggio dal sistema operativo 1.3 a quello 2.0: tanto nella forma, che nella sostanza.
La maggior parte dei comandi del Dos, pur ottimizzati nel loro funzionamento interno, hanno infatti conservato una sintassi d'uso quasi identica rispetto alla precedente versione. Non vi è stata una vera e propria rivoluzione, ma le piccole e grandi modifiche apportate hanno favorito un sempre maggiore comfort dell'utente, e una sempre più accentuata versatilità e affidabilità generale.
Sulla base di questa premessa, resta dunque totalmente valido quando già descritto nel capitolo 6 (il CLI 1.3), mentre alcuni aspetti andranno aggiornati.

Nuove finestre

Primo tra tutti, lo stesso ambiente fisico di lavoro.
La finestra Shell si è infatti adeguata alle generali caratteristiche dell'ambiente 2.0, ed è quindi fornita degli stessi accessori (gadget) presenti nelle altre finestre Workbench: quello unico di profondità all'estremità in alto a destra, quello di zoom immediatamente contiguo

(con relativo refresh e riassestamento dell'eventuale testo presente all'interno della finestra), e quello di chiusura nell'angolo in alto a sinistra.
Grazie a quest'ultimo, non è dunque più obbligatorio ricorrere ai comandi Endcli o EndShell che mantengono comunque la loro validità in rapporto all'uso nel contesto di Batch file (o "file comandi" che dir si voglia).
Va comunque aggiunto che la presenza del gadget di chiusura, come altre caratteristiche delle finestre di lavoro Shell, sono ampiamente modificabili grazie alla sintassi del comando NewShell.
Questo, se digitato direttamente sulla riga di comando (si ricorda che con il Workbench 2.0 è anche disponibile la voce Execute Command diretamente da menu), o se inserito nell'ambito del file Startup-sequence, serve ad aprire na nuova finestra Shell.
La sua sintassi, leggermente ampliata rispetto al vecchio 1.3, può essere così schematizzata:

```
NEWSHELL <finestra> <FROM nomefile>
```

Entrambi i parametri sono facoltativi e, se assenti, il sistema aprirà una finestra standard, cercando comunque di eseguire prima quanto contenuto in un file di nome Shell-Startup memorizzato nella directory S del disco di sistema (quello adoperato per attivare Amiga, tipicamente il Workbench).
Il parametro <finestra> serve appunto a precisare le caratteristiche della Shell: dimensioni, eventuale titolo e opzioni come il gadget di chiusura secondo questo schema:

```
CON:x/y/larghezza/altezza/titolo/opzione
```

Un esempio chiarirà meglio il tutto. Da ambiente Workbench, si prema il tasto Amiga Destro e, mantenendolo abbassato, si prema anche il tasto E. Si attiverà una finestra dotata di riga di comando. Al suo interno, si digiti ora:

```
NEWSHELL CON:0/150/640/100
```

facendo seguire un canonico Return (o Enter). Si aprirà una finestra Shell che occuperà la porzione inferiore dello schermo, priva del gadget di chiusura e senza titolo.
I valori 0 e 150 precisano infatti la posizione del suo angolo superiore sinistro: il primo indica il numero di pixel in senso orizzontale (a partire quindi dal bordo sinistro dello schermo), 150 invece di quanti pixel dal bordo superiore deve essere distante.
Per chiudere la finestra, si impartisca dunque ENDCLI al suo interno dopo avere eventualmente attivato la finestra con un click del mouse, oppure a premere il tasto CTRL e \ (back slash), altro metodo molto sbrigativo per bypassare la digitazione di ENDCLI.
Si provi ora a reimpartire il comando in modo che appaia in questa forma:

```
NEWSHELL CON:0/150/640/100/test/close
```

Stavolta, la nuova finestra avrà il titolo "test" sulla barra superiore e potrà essere richiusa agendo sul gadget in alto a sinistra.
Volendo eliminare il titolo, ma conservando la possibilità di richiudere la finestra con il mouse, la stessa istruzione diventerà

```
NEWSHELL CON:0/150/640/100//close
```

con una doppia barra obliqua a sostituire la descrizione del titolo. Qualora si dimenticasse questo particolare, verrebbe invece aperta una finestra con titolo "close" e senza gadget di chiusura.
Il secondo parametro, FROM <nomefile>, serve invece ad evitare che il sistema esegua, prima dell'apertura di ogni finestra Shell, il file Shell-Startup, costringendolo ad eseguire un nostro file. Opzione, questa, molto comoda, ma i cui risvolti presuppongono una già acquisita padronanza del DOS. Argomento, dunque, più consono alle rubriche specializzate di Amiga Magazine.

La console

Anche l'uso comune dell'ambiente, quello legato all'editing vero e proprio dei comandi, presenta qualche novità. Intanto, vediamo di ricordare di quali opzioni si dispone nella fase di digitazione, oltre ai consueti tasti cursore sinistro/destro per spostarsi nell'ambito della riga di comando:

Shift+freccia sinistra Spostamento a inizio riga
Shift+freccia destra Spostamento a fine riga
Ctrl+W Cancella la parola a sinistra del cursore
Ctrl+X Cancella l'intera riga
Ctrl+K Cancella tutto dal cursore a fine linea
Ctrl+Y Ripristina quanto cancellato con Ctrl+K
Ctrl+U Cancella tutto dal cursore a inizio linea

A queste facilitazioni, va poi aggiunta la già preesistente facoltà di sfruttare la cosiddetta history, ovvero la possibilità di richiamare sulla linea di comando quanto digitato in precedenza, adoperando i tasti cursore alto/basso. In pratica: se, per esempio, nel corso della attuale sessione si è in precedenza adoperata nell'ordine una serie di comandi List, Echo "ciao" e Status, ad ogni pressione del tasto "freccia in alto" si riproporrà sulla riga di comando prima Status, poi Echo "ciao", e infine List. Con il tasto cursore verso il basso, si "avanzerà" invece di nuovo nel precedente ordine.
Questa feature era già implementata anche nella versione 1.3, ma a partire da DOS 2.0 si aggiunge una ulteriore comodità: la ricerca selettiva. In pratica, è sufficiente digitare qualcosa (anche un solo carattere) sulla riga di comando, quindi premere il tasto Shift assieme al tasto cursore verso l'alto. Automaticamente, verrà richiamata qualunque digitazione precedente che inizia con gli stessi caratteri specificati.
Altro esempio: si digiti, all'interno di una finestra Shell, questa serie di comandi, facendo seguire la pressione del Return a ogni riga, ignorando eventuali segnalazioni di errore:

```
Copy
CD
Echo "ciao"
Info
Avail
```

Ora si provi a digitare solo la lettera C, a premere il tasto Shift e, mantenendolo abbassato, il tasto cursore verso l'alto: sulla riga apparirà il comando Cd. Ripetendo l'operazione (con i soli tasti Shift e cursore in alto), apparirà Copy.
Come intuibile, la cosa tornerebbe estremamente utile se, dopo lunghe sedute dossiane, si volesse rintracciare un comando impartito decine di istruzioni prima, magari di sintassi molto lunga e quindi non facile da ricordare a memoria.

Spostarsi tra le directory

Il capitolo 6, per un primo approccio alla struttura generale del DOS di Amiga, dedicava ampio spazio a una delle esigenze primarie nell'eseguire un programma, il corretto posizionamento all'interno di una directory, o di un diverso disco. Allo scopo, era necessario un ricorso massiccio al comando CD, diventato ora del tutto superfluo nell'uso diretto della riga di comando. Ricorriamo ancora ad un esempio: se ancora non avete richiuso la finestra Shell (nel caso, riapritela), la riga di comando riporterà come prompt (quanto appare all'estremità sinistra) il nome della directory corrente, ovvero Workbench2.0 se state adoperando il normale disco di sistema. Supponiamo di volerci spostare in una sua subdirectory, ad esempio Utilities, per vedere cosa c'è dentro.
Con il vecchio 1.3 avremmo dovuto digitare un comando Cd Utilities, e in effetti questo funzionerà correttamente anche in ambito 2.0. Con il nuovo Dos, però, non è più necessario adoperare Cd!
Per constatarlo, si provi a digitare solo Utilities (sempre seguito dalla pressione del Return): il prompt segnalerà l'avvenuto cambiamento di

directory esattamente come se si fosse adoperato Cd, e come verificabile impartendo Dir (o List) per visualizzarne il contenuto. Chiaro che, per un corretto spostamento, varranno sempre le regole associate a quel comando, ovvero la directory di arrivo dovrà comunque essere "visibile" da quella di partenza, o dovrà essere specificato tutto il percorso per giungervi.
Se, quindi, dalla posizione raggiunta nell'esempio (la directory Utilities del disco Workbench) ci si volesse posizionare in quella di nome System, occorrerebbe impartire Workbench2.0:System, o più semplicemente Sys:system.

I comandi interni

Adoperare la finestra Shell significa, genericamente, impartire precisi comandi seguendo la sintassi loro assegnata. Ma cosa si intende per comando DOS? Come già accennato nel capitolo 6, si tratta di programmi che, una volta invocati, verranno prelevati da particolari directory del dischetto (la più comune è quella di nome C) ed eseguiti, proprio come qualunque altro. O almeno, questo valeva in assoluto quando si adoperava il DOS 1.3. Ebbene, con il 2.0 non è sempre così, e questo porta a vantaggi davvero notevoli, come si capirà tra breve.
Nell'esempio appena riportato, si accennava al comando CD: non più indispensabile nelle digitazioni dirette (ma sempre opportuno nell'ambito di batch file, non foss'altro che per compatibilità all'indietro), ma comunque possibile. Se, però, si prova ad impartire List C:, non si troverà traccia di questo comando all'interno di quella directory, e lo stesso dicasi per molti altri prima adoperati: NewShell ed Endcli, per esempio. Eppure, visto che hanno funzionato, da qualche parte dovranno pur trovarsi.
Per rintracciarli, si digiti il comando Resident all'interno della finestra Shell: eccoli elencati, assieme a una gran quantità di altri. Il DOS 2.0, dunque, contiene quella serie di comandi già nelle sue memorie di sistema, con due grossi vantaggi: la velocità di esecuzione, e la possibilità

di adoperarli indipendentemente dal floppy inserito nel drive principale, evitando quella noiosa sequela che obbligava a continui cambi di dischetto gli utenti sprovvisti di hard disk o di secondo drive.
Per constatarlo, si estragga dal drive il floppy Workbench con il quale si è lanciato Amiga, e si digiti all'interno della finestra Shell un comando Ask "Premi Return": la stringa Premi Return verrà stampata sulla riga sottostante, e il computer si porrà in stato di attesa fino a che l'operazione richiesta non sarà stata compiuta.
Senza soffermarci sulle modalità d'uso del comando Ask, lo si è in pratica visto in azione senza la presenza del disco di boot!
Come intuibile, la cosa può risultare di grande comodità non solo quando ci si limita alla riga di comando, ma soprattutto nell'ambito di batch file che, tra l'altro, risulteranno molto più veloci grazie all'uso dei comandi interni. Grazie a loro, inoltre, non sono più necessarie tutte quelle complesse operazioni "succhiamemoria" che obbligavano gli utenti con Amiga in versione base a copiare moltissimi comandi in RAM Disk, per averli disponibili quando nell'unico drive si doveva inserire un altro floppy.

Facili scritture

Tra i molti altri aspetti degni di nota, va infine citato un elemento che ha subìto un enorme passo avanti: l'editor di testi ASCII fornito assieme al sistema Ed. Pur rimanendo uno strumento di emergenza e non paragonabile a veri strumenti di scrittura, si è tuttavia evoluto divenendo utile quando si deve, per esempio, elaborare o modificare in fretta un file batch come la Startup-Sequence. Non disponendo di hard disk, il ricorso a un editor esterno richiederebbe una discreta perdita di tempo e varie operazioni supplementari, che con Ed possono esere evitate.
Per attivarlo è sufficiente digitare Ed <nomefile> sulla linea di comando Shell, adoperando, per esempio, RAM:test per definire un file temporaneo.
La novità più rilevante è legata alla disponibilità di veri menu sulla barra di schermo, che svolgono le più importanti funzioni di editing, comunque anche associate a sequenze di tasti: Load e Save tramite i requester interni

di sistema (altra grossa novità del DOS 2.0), funzioni di ricerca, e i più basilari spostamenti a inizio/fine documento.

La possibilità di utilizzare dei menu non è molto, ma è decisamente meglio che districarsi nell'infinità di comandi poco intuitivi cui costringeva Ed nelle precedenti versioni.

Capitolo 10

IL WORKBENCH 3.0: ULTIME NOVITA'

Dopo i drastici ed evidenti cambiamenti introdotti dalla Commodore nel passaggio dal sistema operativo 1.3 a quello 2.0, l'avvento dei modelli Amiga 1200 e 4000 ha segnato un ancora più deciso e rivoluzionario passo avanti nella tecnologia informatica. Certamente più radicale del precedente a livello di prestazioni generali e di architettura hardware, ma tutto sommato con minore necessità di adattamento da parte dell'utente già abituato al precedente Workbench.

Lingua nazionale

L'interfaccia fornita dal Workbench si presenta, infatti, in tutto e per tutto simile alla precedente. Stesso look e stessi menu, ma con almeno un'immediata e percebile differenza: la possibilità di leggere in italiano sia le opzioni di questi ultimi, che le segnalazioni di Amiga. Così i Requester diventano Richieste di sistema, e il loro contenuto potrà essere facilmente capito anche da chi non fosse particolarmente ferrato nelle lingue straniere. Tutto ciò grazie alla cosidetta "localizzazione", settabile in modo intuitivo attivando l'icona Locale presente all'interno del cassetto Prefs, e selezionando la lingua preferita con un semplice click del mouse.

Colori

Già nel capitolo 8 ci siamo ampiamente soffermati sulle possibilità offerte dal cassetto Prefs e, a un primo esame superficiale, quasi tutti i

programmi presenti nella versione 3.0 della directory sembrano accedere alle stesse risorse. In molti casi è proprio così, ma con quell'inevitabile qualcosa in più legato alle maggiori potenzialità generali del sistema.
Senza soffermarci troppo su aspetti tecnici che hanno trovato e trovano maggiore spazio nelle rubriche di Amiga Magazine, basterà citare ancora come primo elemento la scelta del numero di colori assegnabili allo schermo Workbench con il programma ScreenMode: dai 16 che già rappresentavano un notevole progresso nel 2.0; ora il solito cursorino raggiunge il limite di 256!
Anche la regolazione della palette (programma omonimo), adeguandosi alla nuova varietà disponibile, può essere regolata sia in modo tradizionale, che agendo su un pointer all'interno di una ruota contenente tutte le componenti cromatiche.

Amiga e MS-DOS

Ancora più che nella precedente versione, il Workbench permette inoltre di operare alcune scelte di sistema senza la necessità di ricorrere a finestre DOS, ne intervenire sulla startup-sequence o su altri file prima di manipolazione pressoché obbligata. Il cassetto Devs, infatti, contiene tutta una serie di subdirectory che condizionano l'attivazione di particolari risorse: basterà trasferire all'interno di una di esse quanto desiderato, per ritrovarselo attivo nell'uso comune sin dallo start.
L'esempio di più immediata utilità è rappresentato dalla possibile utilizzazione di floppy in formato MS-DOS. Con il Sistema Operativo 3.0 (ma già lo consente l'upgrade 2.1) si può tranquillamente infilare nei drive un disco formattato su quella piattaforma e agire su di esso direttamente da Workbench, come si farebbe con qualunque altro floppy Amiga: trascinamento di icone per operazioni di copia, eventuali cancellazioni, o addirittura formattazione in quel tipo di standard. Come intuibile, il trasferimento di dati da/verso i cosiddetti PC diviene, così, estremamente semplice.
Per attivare questa facoltà, può essere sufficiente cliccare su una delle due icone PC0 e PC1 presenti nel cassetto Storage, subdirectory DosDrivers, a seconda che si desideri abilitare alla lettura MS-DOS il drive interno oppure l'unità esterna (DF1:). Già questo rappresenta un

notevole passo avanti: niente modifiche della cosiddetta Mountlist, nessun comando da impartire all'interno di finestre Shell.
Se si volesse poi mantenere permanente questa caratteristica, ovvero fare in modo che si attivi automaticamente dopo ogni reset del computer, sarà sufficiente "trascinare" con il mouse l'icona PC0 (o PC1) nell'omonima directory DosDrivers presente all'interno del cassetto Devs.

Monitor

Stesso tipo di operazione per impostare una modalità video in rapporto al monitor di cui si dispone. Il sistema 3.0, come certamente noto, è infatti in grado di supportare una infinità di risoluzioni grafiche. Per la loro selezione, si dovrà ricorrere ancora a uno dei programmi presenti in Prefs (ScreenMode), ma perché alcune di esse siano realmente utilizzabili sarà necessario possedere monitor adeguati, come i Multisync Commodore 1960, 1940 o 1942.
Per accedere tramite ScreenMode alle modalità di visualizzazione consentite, occorrerà prima scegliere uno dei Monitor elencati nella directory omonima (Monitors). Anche in questo caso, l'attivazione passerà per un un banale trasferimento della relativa icona nel cassetto Devs/Monitors.

MultiView

Multiview è il nome di un programma che viene fornito assieme ai dischi di sistema 3.0 ed è rintracciabile nella directory Utilities. Va citato perché rappresenta una svolta nel modo di visualizzare file del più vario formato. In precedenza, tutto ciò di cui si disponeva era More, utile solo nella lettura di testi e decisamente obsoleto, tanto che programmi di Pubblico Dominio come MuchMore e similari, lo avevano del tutto soppiantato.
Con il nuovo Workbench quel file è scomparso per fare posto a questo nuovo presidio di estrema versatilità. In pratica, MultiView può essere utilizzato anche per la lettura di testi, ma, con eguale facilità, può visualizzare schermate grafiche IFF, riprodurre campionamenti audio dello stesso formato, e dare accesso a un nuovo modo di intendere

l'help. Grazie allo standard AmigaGuide, testi di aiuto su qualunque programma ne sia fornito, possono infatti essere rappresentati, in cosiddetto ipertesto:, vari menu cliccabili con il mouse, o anche specifici termini nell'ambito del testo stesso che, se cliccati, daranno accesso a ulteriori e più dettagliate spiegazioni del termine stesso.
In generale, MultiView può accettare qualunque formato sia presente nella directory DataTypes all'interno del solito cassetto Devs. Per cui, se nuovi formati dovessero essere resi utilizzabili, basterà portarne l'icona al suo interno per consentirne l'utilizzo con MultiView.
Il programma può essere utilizzato in modo normale, ovvero bicliccando sulla sua icona e scegliendo poi il file da visualizzare da requester, oppure cliccando una volta sulla sua icona e, mantenendo abbassato il tasto Shift (maiuscolo), agendo due volte con il pulsante del mouse sull'icona del file da visualizzare. I più esperti, potranno poi far sì che l'attivazione dell'icona di un file di testo, grafico, sonoro o in formato AmigaGuide, porti automaticamente a richiamare MultiView.
Allo scopo occorrerà innanzitutto cliccare una volta sull'icona del file, per esempio, di testo e selezionare Information dal menu Icons del Workbench (Informazioni dal menu Icone se è settata la lingua italiana). Si aprirà una finestra contenente anche un riquadro "Programma associato" (Default tool): se al suo interno si digita MultiView (oppure Workbench:utilities/multiview) e si seleziona poi l'opzione Salva, ogni volta che si biclicchera sull'icona si provocherà la visualizzazione del file tramite MultiView.

Capitolo 11

NON SOLO GAMES

Dopo esserci soffermati a grandi linee sugli aspetti pratici che possono interessare un utente di Amiga, resta ancora inesposto un quesito solo apparentemente banale: cosa farsene di un computer di questa classe? E' infatti luogo comune definire Amiga una perfetta macchina da gioco, non completamente a torto: grazie alle loro eccezionali risorse grafiche e sonore, anche modelli di basso costo come il 500, 600 e soprattutto il 1200 e successori, nel settore ludico risultano pressoché imbattibili.
Ma esistono altri vantaggi: oltre che a giocare, ci si può fare praticamente di tutto, dalle più comuni applicazioni da home computer al superprofessionale, in quest'ultimo caso con l'eventuale supporto di mostri di classe superiore come Amiga 4000. Naturalmente con l'ausilio di adeguato software o hardware addizionale, la cui disponibilità sul mercato è ormai di proporzioni a dir poco smisurate. Tentiamone una sintetica panoramica.

Archivi e calcoli

Database e fogli di calcolo (spreadsheet) sono, di solito, i più gettonati elementi software di un computer. Non è forse il caso di Amiga, che vede nella grafica e nel multimediale il suo principale motivo di successo, ma la velocità di elaborazione già nei modelli di base non poteva far mancare questo tipo di utilizzazione agli utenti anche non professionali. Seppure non ricco come altri, questo settore vede imperare da ormai molto tempo Superbase tra i programmi di archiviazione, ormai divenuto Superbase 4, evolutosi fino a raggiungere una compatibilità che va oltre la categoria Amiga, grazie anche a una sua implementazione esistente su piattaforma MS-DOS. A questo si affiancano comunque, giusto per una

citazione, altri "pezzi" storici quali Organize!, Pen Pal e MicroFiche Plus. Piuttosto limitato il settore dei fogli elettronici, anch'esso da tempo dominato da capisaldi come MaxiPlan e Advantage, quest'ultimo in grado di importare il formato file dei blasonati Lotus ed Excel dell'universo MS-DOS.

Trattamento dei testi

Di tutte, è forse l'applicazione più usuale: può sempre capitare di dover scrivere una lettera, una lista, o un documento qualunque. Con l'aiuto di una stampante, Amiga è in grado di facilitare molto questo compito, al pari e forse meglio di altri tipi di computer. La maggior parte dei programmi adatti a questo scopo, i cosiddetti word processor, permettono, infatti, non solo di stilare e formattare testi come più aggrada, ma anche, volendo, di inserire nel documento immagini grafiche, giungendo molto vicino a quanto ottenibile con programmi specializzati nel cosiddetto DTP (si veda più avanti). La scelta del software più adatto è molto soggettiva, ma tra tutti spiccano nomi come ProWrite, Final Copy II, Wordworth, tutti dotati delle caratteristiche appena citate. Se, poi, si preferisse qualcosa di meno "grafico", ma particolarmente rivolto all'uso della ligua italiana, allora Cloanto C1-Text non trova rivali.

DTP

Sigla che sta per Desk Top Publishing, identificabile con la realizzazione di sofisticati elementi grafico-testuali su carta, con vere e proprie impaginazioni come quelle riscontrabili su quotidiani, riviste, depliant, eccetera. In questo settore, tra l'altro, Amiga può essere sfruttato anche professionalmente, grazie a presidi come Professional Page e Page Stream, due colossi in grado di generare, oltre alla stampa, dei file PostScript adoperabili nei processi di elaborazione tipografica. Inutile dire che il trattamento riguarderà sia i testi che la parte grafica; nel caso del DTP è rappresentata in modo cosiddetto Vettoriale, con la possibilità di modificarne le dimensioni senza perdita di qualità con il variare della risoluzione. Proprio per questa esigenza, oltre che agli strumenti interni ai due programmi, può risultare utile affiancare loro un software dedicato alla grafica vettoriale, come Professional Draw o ProVector. Le immagini create con questo editor, comunemente chiamate clip, possono poi

essere importate sia da Professional Page che da Page Stream.
Questo tipo di applicazione, per quanto teoricamente possibile su tutti i modelli di Amiga, necessita di abbondanti risorse in termini di memoria RAM, e risente pesantemente della velocità del sistema: per usi "seri", sono dunque consigliabili Amiga di categoria superiore o comunque accelerati (oltre che espansi).

Trattamento della grafica

E siamo al "clou". La grafica, su Amiga, ha sempre avuto una netta predominanza su tutti gli altri applicativi, arricchendosi di sempre nuovo software dagli effetti strabilianti. Dai primi passi mossi dal tuttora imperante Deluxe Paint e consimili per un semplice (si fa per dire) editing di immagini e animazioni, il panorama comprende ora tutta una serie di supporti in grado di modellare la grafica con estrema versatilità e con sempre maggiore automatismo.
Le manipolazioni possono essere innumerevoli. Per esempio, la resa tridimensionale di un oggetto, con effetti di luce annessi, consentita da programmi come il vecchio Sculpt o i più potenti e recenti Imagine e Real 3D. O, ancora, la creazione dei più incredibili effetti di distorsione, bassorilievo o trasformazione progressiva di un'immagine in un'altra, resi possibili da Art Department Professional e Morphplus, Imagemaster, ImageFx e il suo Cinemorph.
Per non parlare dell'ormai diffuso supporto della grafica "che più nitida non si può", ovvero quella a 24 bit (16 milioni di colori), resa concretamente operativa da schede di prestazioni professionali come Rambrandt, OpalVision, ImpactVision, Harlequin, e altre ancora. Schede che, tra l'altro, per le loro varie possibilità di utilizzo, ben si prestano ad un uso rivolto al DTV. Anche in questo settore, come intuibile, la configurazione hardware riveste una notevole importanza, con il quasi obbligo di processori veloci, coprocessori matematici e molta RAM addizionale con il crescere delle prestazioni volute.

Multimediale e DTV

E' questa la categoria in cui Amiga è diventata insuperabile, soprattutto nel rapporto prezzo/prestazioni. La realizzazione di presentazioni composte di grafica e sonoro, animazioni e testi, facilmente riversabili su

supporti magnetici video anche con il solo ausilio di un genlock, è resa comoda e facile da tutta una serie di prodotti che sfruttano appieno le risorse del computer. Quella che un tempo veniva definita semplice "titolazione", ha raggiunto livelli di professionalità incredibile, e la serie di programmi dedicati al multimediale è divenuta piuttosto lunga: al commodoriano Amiga Vision, si aggiungono infatti presidi di vario livello e costo come CanDo, The Director, Scala Multimedia, Hyperbook, Deluxe Video, e vari altri. Tutti in grado di importare e collegare in un'unica unità eseguibile, con varia versatilità, file sonori campionati o sotto forma di moduli musicali, file grafici o animazioni per lo più in formato IFF e ANIM, e testi ASCII.

Suono e musica

Fra tanta grafica, non va dimenticato infine il suono, che su Amiga assume un'importanza rilevante grazie alle sue risorse interne, ma soprattutto grazie alla nutrita schiera di economici supporti hardware e software in grado di acquisire campionamenti ed elaborarli, o anche sommarli in unità musicali (moduli) complesse. Nel primo caso, hardware tradizionale di uso per così dire domestico trova in Amas 2 un supporto sicuramente valido con possibilità di stereo e supporto MIDI. Ma le nuove frontiere parlano di digitalizzazione a 16 bit, e non si può dunque non citare una scheda come SunRize, di ben più elevate prestazioni (e relativi costi). Per normali esigenze di editing e campionamento a 8 bit, da un punto di vista software fanno la parte del leone presidi come AudioMaster 4, Audition 4 e A-Sound. I suoni digitalizzati con tali strumenti possono poi essere eventualmente utilizzati, se è il caso, dopo l'editing, per creare brani mediante programmi che, di solito, richiedono una certa conoscenza di base della notazione musicale: i vari Noise Tracker, Pro Tracker e Sound Tracker sfruttano i normali 4 canali audio di Amiga. L'ultima generazione di questo titpo di software è però in grado di adoperare anche otto canali e un input semplificato, come avviene in Octamed. Anche i non esperti, comunque, possono ricorrere a un software che non richiede conoscenze approfondite: per esempio, Quartet, Hyperchord, o Music Mouse. Non mancano poi dei veri supporti professionali, capaci di sfruttare appieno un interfacciamento MIDI. Per fare qualche nome: il classico KCS, cui si affiancano Midi Quest e Xor.

GLOSSARIO

AA
Detto anche AGA, è il nuovo chipset montato su 1200 e 4000. Permette schermi a 256 colori e il modo HAM8. La palette è a 24 bit (1.6 milioni di colori).

Alias
Comando CLI che permette di usare abbreviazioni invece dei nomi standard.

Agnus
Uno dei chip custom Amiga: si occupa della gestione della Chip RAM e contiene il Blitter.

AmigaDOS
Quella parte del sistema operativo che si occupa dell'archiviazione dei dati su disco. Su Amiga funziona in multitasking e gestisce File System con struttura ad albero.

ARexx
Linguaggio interpretato parte integrante del sistema operativo Amiga a partire dal 2.0

ASCII
Tabella standard che associa ad ogni carattere dell'alfabeto un valore numerico compreso grossomodo tra 0 e 127.

Assign
Il comando che permette di assegnare un nome scelto dall'utente a delle directory (o file) posti su disco. Le directory create vengono dette logiche o device logici.

Binario
Sistema di numerazione in base 2.

Bit
Informazione minima disponibile in un sistema digitale. Può assumere due soli valori: acceso o spento, 1 o 0, alto o basso, concetti in pratica equivalenti. La matematica binaria è il sistema di notazione più comodo per rappresentare dati in questa forma.

Bitmap
Il formato standard per la gestione delle immagini su Amiga sia in memoria che su file. Una bitmap può essere formata da 1 o più bitplane.

Bitplane
Piano di bit, contiene un bit per ogni pixel sullo schermo. Ad ogni pixel possono corrispondere più bit appartenenti a diversi bitplane che, combinati tra loro, formano un numero che indica il colore di quel pixel.

Blitter
Coprocessore grafico Amiga votato alla gestione di immagini in formato bitmap.

BPS
Bit al secondo. Misura la velocità di porte seriale, modem. Certe volte indica i baud al secondo.

Buffer
Zona di memoria utilizzata per accumulare dati prima di inviarli in uscita verso una porta esterna (parallela, seriale, floppy, SCSI...).

Bug
Errore presente in un programma.

Bridgeboard
Scheda Zorro detta anche Janus contenente una scheda madre IBM compatibile. Si può montare su A2000, A3000 e A4000.

Byte
Insieme di 8 bit.
Può assumere 256 valori differenti dovuti alle possibili combinazioni dei singoli bit che lo compongono. A queste 256 combinazioni vengono per convenzione attribuiti i valori 0-255.

Cache
Memoria molto veloce utilizzata da 68020, 68030 e 68040 per limitare gli accessi alla più lenta RAM di sistema.

CD-ROM
Sistema di archiviazione di tipo ottico (laser) analogo quello usato dai CD audio. Con questi ultimi condivide anche il formato fisico, per cui i lettori di CD-ROM sono spesso in grado di leggere i CD audio.

CD-Photo
Nuovo sistema di archiviazione di dati grafici (fotografie) sviluppato dalla Kodak. Usa lo stesso formato fisico dei CD audio e dei CD-ROM.

CLI
Interfaccia a comandi Amiga. Utilizza la tastiera per ricevere input dall'utente. La Shell costituisce una versione migliorata.

Compressione
Operazione con la quale si riduce l'occupazione di memoria di programmi, immagini, campioni audio o qualsiasi altra cosa. Al momento dell'uso occorre effettuare l'operazione inversa.

CPS
Caratteri al secondo. utilizzando per misurare la velocità delle stampanti, ad esempio.

Decimale
Sistema di numerazione in base 10.

Device
Dispositivo. Su Amiga è un programma cui il sistema demanda la gestione di compiti particolari come l'audio, il mouse, il calcolo del tempo, il floppy, la porta IDE, SCSI, seriale, parallela e così via. Il termine può essere usato anche per indicare gli handler del DOS (come DF0:), mentre con il termine di device logico si possono indicare le directory logiche create mediante Assign.

Digitalizzatore
Dispositivo capace di convertire un segnale analogico in informazioni digitali. Usato spesso per convertire segnali audio o video in dati gestibili da un computer.

DPI
Punti per pollice. misura la risoluzione di scanner e stampanti.

Drawer
Cassetto. Elemento del Workbench che rappresenta una directory o un volume.

ECS
Il chip set grafico che ha fatto la sua comparsa con il 3000, il 500 Plus e il 600. Permette l'uso di 1 o 2 MB di RAM più vasta e nuovi schermi di tipo VGA.

Esadecimale
Sistema di numerazione in base 16.

Exec
Il cuore del sistema operativo Amiga. Gestisce la memoria e il multitasking.

File
Elemento di un sistema d'archiviazione, di solito dotato di struttura gerarchica (ad albero). I file costituiscono le foglie, cioè gli elementi terminali, della struttura. Le directory indicano gli elementi non terminali (i rami da cui dipendono più foglie). File può essere utilizzato anche per indicare un elemento generico del sistema di archiviazione (sia ramo che foglia).

File System
Sistema di archiviazione dotato, su Amiga, di struttura gerarchica, ad albero, composta da rami (directory) da cui dipendono altri rami o foglie, cioè elementi terminali (file). In Amiga è gestito da AmigaDOS mediante gli handler.

Finestra
Window, in inglese. Indica quella parte dello schermo in cui compaiono gadget, menu. Uno schermo può contenere più finestre, le quali possono essere spostate, sovrapposte, ridimensionate.

Floptical
Sistema di archiviazione su dischi removibili in tecnologia ottica e magnetica assieme. Usa dischi del medesimo formato dei floppy da 3.5.

Floppy
Sistema di archiviazione su dischi removibili di tipo magnetico. Su Amiga si usano soprattutto quelli a doppia faccia doppia densità da 3.5 pollici. La capienza disponibile è di 880 KB. Con il 4000 sono apparsi floppy ad alta densità da 3.5" (distinguibili per la presenza di due fori sul dischetto) capaci di 1.76 MB.

Formattare
Preparare un disco vergine perché possa ricevere i dati di un determinato sistema operativo. Un disco appena formattato è da punto di vista utente completamente vuoto, in verità sono presenti in esso una serie di dati che permettono al sistema operativo di riconoscerlo e utilizzarlo.

FPS
Fotogrammi al secondo. Misura la velocità di animazioni e filmati in genere.

Gadget
Un pulsante (o altro) che può essere selezionato con il pulsante sinistro del mouse sotto Intuition.

Genlock
Dispositivo hardware che permette di miscelare il segnale video RGB con un segnale videocomposito o S-VHS.

Gigabyte
Esattamente 2 elevato alla trentesima, pari a 1.073.741.824 byte.

HAM
Particolare schermo Amiga che permette 4096 colori con l'ECS e 262144 colori (in realtà 16 Milioni) con l'AA. I file HAM possono avere 6 (HAM) o 8 bitplane (HAM8).

Handler
Programma di gestione del file system. Fanno parte dell'AmigaDOS che li utilizza per gestire i vari dischi (RAM:, DF0:, HD0:...), Su Amiga funzionano in multitasking.

Hard disk
Sistema di archiviazione di tipo magnetico, capace di grandi velocità e di dimensioni che variano da poche decine di Megabyte a mille e più (Giga).

Hires
Alta risoluzione: prevede schermi da 640x256 pixel in PAL e da 640x512 in modo interlacciato.

Icona
Immagine che rappresenta file, directory,

programmi. Contiene dati utili al programma da eseguire come eventuali parametri. In Amiga sono contenuti su file separati con il suffisso .info.

IDE
Interfaccia per hard disk diffusa soprattutto su sistemi MS-DOS. La montano di serie anche Amiga 600, 1200 e 4000. Permette il collegamento di 1 o 2 hard disk.

Input
Ingresso. Si contrappone ad output.

Interlacciato
Detto anche interallacciato, lace, interlace, indica schermi a 25 frame al secondo che permettono di elevare la risoluzione verticale da 256 a 512 pixel, a costo di un fastidioso sfarfallio. I segnali video (TV, videoregistratore) sono sempre interlacciati.

Intuition
Parte del sistema operativo Amiga che si occupa dell'interfaccia utente composta da schermi, finestre, icone, gadget, menu.

Janus
Vedasi Bridgeboard.

Joystick
Dispositivo hardware che permette di muovere un oggetto sullo schermo e di attivare altre opzioni mediante il pulsante di fuoco. Utilizzato soprattutto nei videogiochi. Si collega alla seconda porta joystick di Amiga.

Jumper
Ponticello. Interruttore o deviatore hardware utilizzato spesso per configurare schede d'espansione.

Keyword
Parola chiave. Parametro per comandi CLI.

Kickstart
Parte principale del sistema operativo Amiga.
Contenuto di solito nelle ROM da 256 KB o da 512 KB (2.0 e superiori). Le versioni più utilizzate e diffuse sono la 33 (1.2), 34 (1.3), 37 (2.0), 39 (3.0).

Longword
Insieme di 2 word, 4 byte, 32 bit. Può assumere i valori 0-4294967296.

Lores
Bassa risoluzione: in PAL corrisponde a uno schermo da 320x256 pixel o 320x512 (Lores interlacciato).

Megabit
Esattamente 2 elevato alla ventesima, pari a 1048576 bit. Abbreviato con Mb, corrisponde a 131072 byte.

Megabyte
Esattamente 2 elevato alla ventesima, pari a 1048576 byte. Abbreviato con MB.

Menu
Parte dell'interfaccia grafica che permette di impartire comandi o selezionare opzioni. Su Amiga è associato alle finestre ed è normalmente implementato nella versione a tendina (a discesa). Si attiva col pulsante destro del mouse.

Modem
Dispositivo hardware che trasforma dati digitali proveniente da un'interfaccia seriale in un segnale che si può inviare su una linea telefonica standard o dedicata.

Mouse
Dispositivo hardware che consente di muovere un puntatore sullo schermo e di selezionare mediante i due pulsanti i menu o qualsiasi altro punto dello schermo. Su Amiga si utilizza normalmente il pulsante destro per i menu e quello sinistro per tutto il resto. Si collega alla prima porta joystick di Amiga.

Multiscan
Si dice di monitor che permette di agganciare segnali con diverse frequenze orizzontali. Con Amiga 1200 e 4000 è preferibile l'uso di un monitor di questo tipo (frequenze 15KHz-32KHz). Viene anche detto Multisync.

NTSC
Il segnale video utilizzato in USA: 15 KHz di frequenza orizzontale e 60 Hz di frequenza verticale.
Corrisponde su Amiga a schermi da 320 o 640 colonne per 200 o 400 (interlacciato) righe.

Output
Uscita. Si contrappone a input.

Overscan
La parte dell'immagine video che circonda l'area di uno schermo Amiga standard.

PAL
Lo standard televisivo utilizzato in Italia: frequenza orizzontale 15 KHz e verticale 50 Hz. Corrisponde su Amiga a schermi da 320 o 640 colonne per 256 o 512 (interlacciato) righe.

Parallela
Interfaccia standard per il trasferimento di dati tra dispositivi hardware.
E' molto utilizzato per il collegamento al computer di stampanti e digitalizzatori audio

o video.

Pixel
Un singolo punto luminoso dello schermo.

Ponticello
Vedasi Jumper.

Porta
Canale di comunicazione fra il computer e il mondo esterno.
Su Amiga sono utilizzate soprattutto la porta parallela, seriale, joystick, mouse, SCSI, IDE, floppy, PCMCIA.

Preferences
L'insieme delle preferenze dell'utente riguardanti l'ambiente Amiga. Sono contenute in file su disco e gestite mediante appositi comandi di sistema.

Protocollo
Insieme di regole che definiscono uno standard per la comunicazione fra dispositivi hardware o programmi.

PCMCIA
Interfaccia standard a 16 bit presente su A600 e A1200 per il collegamento a un computer di espansioni di memoria o altre periferiche.

RAM
Memoria volatile costituita da chip di vario formato (ZIP, SIMM, DIP) utilizzata per conservare dati e programmi. Allo spegnimento della macchina, i dati in memoria RAM vanno perduti. La velocità della RAM si misura in nanosecondi (ns). Su Amiga si usano RAM che variano tra i 150 e 40 ns.

RGB
Il tipo di segnale prodotto dall'uscita video di un computer. Quello Amiga è a 15 KHz.

Redirezione
Inviare l'input o l'output di un comando CLI verso una destinazione diversa da quella di default.
E' espresso dai caratteri ">" (output) o "<" (input).

Risoluzione
Numero di punti che compongono uno schermo (o altro).
Si esprime con due valori: larghezza e altezza.

ROM
Memoria non volatile che su Amiga contiene buona parte del sistema operativo.
Nei primi Amiga era da 256 KB, a partire dal 2.0 si è passati a 512 KB. I sistemi più recenti sono già predisposto per future ROM da 1 Megabyte.

Schermo
Area visibile e utilizzabile dell'immagine video.
Amiga è un sistema multischermo: più schermi con risoluzioni diverse possono apparire, sovrapporsi, scorrere l'uno sull'altro a video.

SCSI
Small Computer System Interface. Interfaccia standard per il collegamento di periferiche a personal computer. Permette di collegare fino a 7 periferiche.

Seek
Spostamento della testina (reale o immaginaria) da un punto all'altro di un file o di un disco.

Seriale
Interfaccia standard per la comunicazione di dati fra dispositivi hardware diversi. Su Amiga segue lo standard RS-232. Usata soprattutto per collegare modem e stampanti.

Shell
Versione migliorata del CLI.

Switch
Interruttore. Un tipo di parametro per i comandi CLI.

Tool Type
I parametri che il Workbench comunica a un programma al momento del lancio. Sono contenuti nelle icone.

Videocomposito
Il tipo di segnale utilizzato dalle TV e dai videoregistratori: 15 Khz di frequenza orizzontale e 50 Hz di frequenza verticale in PAL.

VGA
Scheda grafica per sistemi MS-DOS con frequenza orizzontale a 30 KHz circa. I monitor VGA e SVGA permettono di visualizzare solo alcuni dei modi grafici disponibili su A1200 e A4000.

Word
Insieme di due byte, 16 bit. Può assumere i valori 0-65535.

Workbench
Programma di gestione file, dischi e applicativi con interfaccia a icone.

Zorro
Standard utilizzato dagli slot d'espansione Amiga.
Sul 2000 è presente lo standard Zorro II a 16 bit, Su 3000 e 4000, lo Zorro III (compatibile con lo Zorro II) a 32 bit.